FortiManager - XML API Reference

VERSION 5.2.4

FORTINET DOCUMENT LIBRARY

http://docs.fortinet.com

FORTINET VIDEO GUIDE

http://video.fortinet.com

FORTINET BLOG

https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT

https://support.fortinet.com

FORTIGATE COOKBOOK

http://cookbook.fortinet.com

FORTINET TRAINING SERVICES

http://www.fortinet.com/training

FORTIGUARD CENTER

http://www.fortiguard.com

END USER LICENSE AGREEMENT

http://www.fortinet.com/doc/legal/EULA.pdf

FEEDBACK

Email: techdocs@fortinet.com

September 23, 2015

FortiManager 5.2.4 XML API Reference

05-524-293619-20150923

TABLE OF CONTENTS

Change Log

Date	Change Description
2015-09-23	Initial release.

Introduction

FortiManager 5.2 includes a web services interface that facilitates integration with provision systems.

This guide describes how to use the XML-based FortiManager Application Programming Interface (API) to obtain information from the FortiManager unit, run scripts to modify device configurations, and install modified configurations to managed devices.

What’s New

What's New in FortiManager 5.2.4

No elements were added or removed in 5.2.4.

What's New in FortiManager 5.2.3

No elements were added or removed in 5.2.3.

What's New in FortiManager 5.2.2

The following elements have been added in FortiManager 5.2.2:

- addPolicyPackage
- assignGlobalPolicy

What's New in FortiManager 5.2.1

The following element changes have been made in FortiManager 5.2.1:

- runFazReport
- Add filters to generate per user reports.

Using the FortiManager API

The FortiManager enables you to configure managed FortiGate devices through a web services interface.

This sections includes the following topics:

Connecting to FortiManager web services

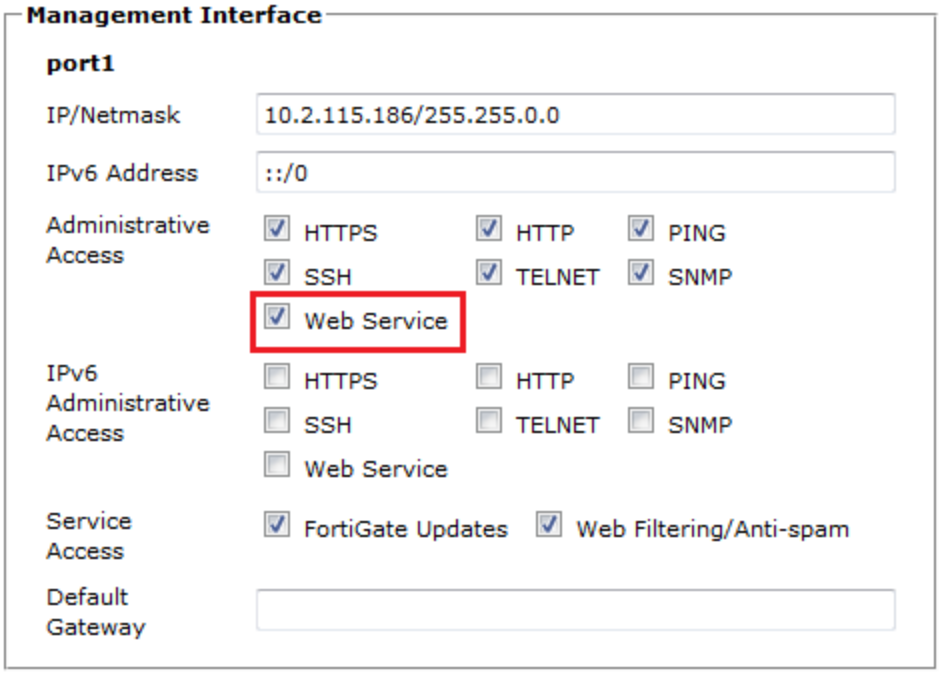
To start working with web services of your FortiManager device, you must enable web services, and download the Web Services Description Language (WDSL) file. The WDSL file defines the XML requests that you can make and the responses the FortiManager system can provide.

Enabling web services

Web services must be enabled on the network interface that the client will connect to.

To enable web services on an interface with the GUI:

1. Go to *System Settings* > *Network* > *Interface*.
2. Select the *Edit* icon for the interface that you need to enable web services on.



3. In the *Administrative Access* section, select *Web Service.*
4. Select *OK* to apply the changes.

To enable web services on an interface using the CLI

Enter the following command line interface (CLI) commands:

```
config system interface
   edit <port>
      set allowaccess webservice
   end
end
```

where `<port>` is the network interface that you want to use for web services.

The `allowaccess` command should also include the other types of administrative access that you want to permit. For example, to allow HTTPS, SSH, and Web Services, enter the CLI command `set allowaccess https ssh webservice`.

The FortiManager unit handles web services requests on port 8080.

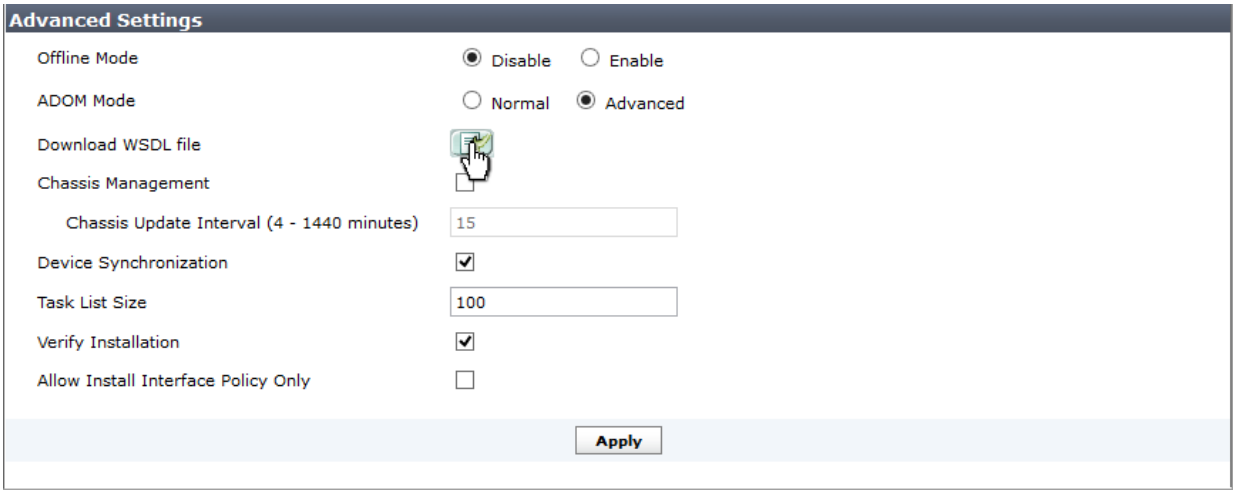
Obtaining the WSDL file

You can download the WDSL file from the GUI, or directly from your FortiManager at the following URL:

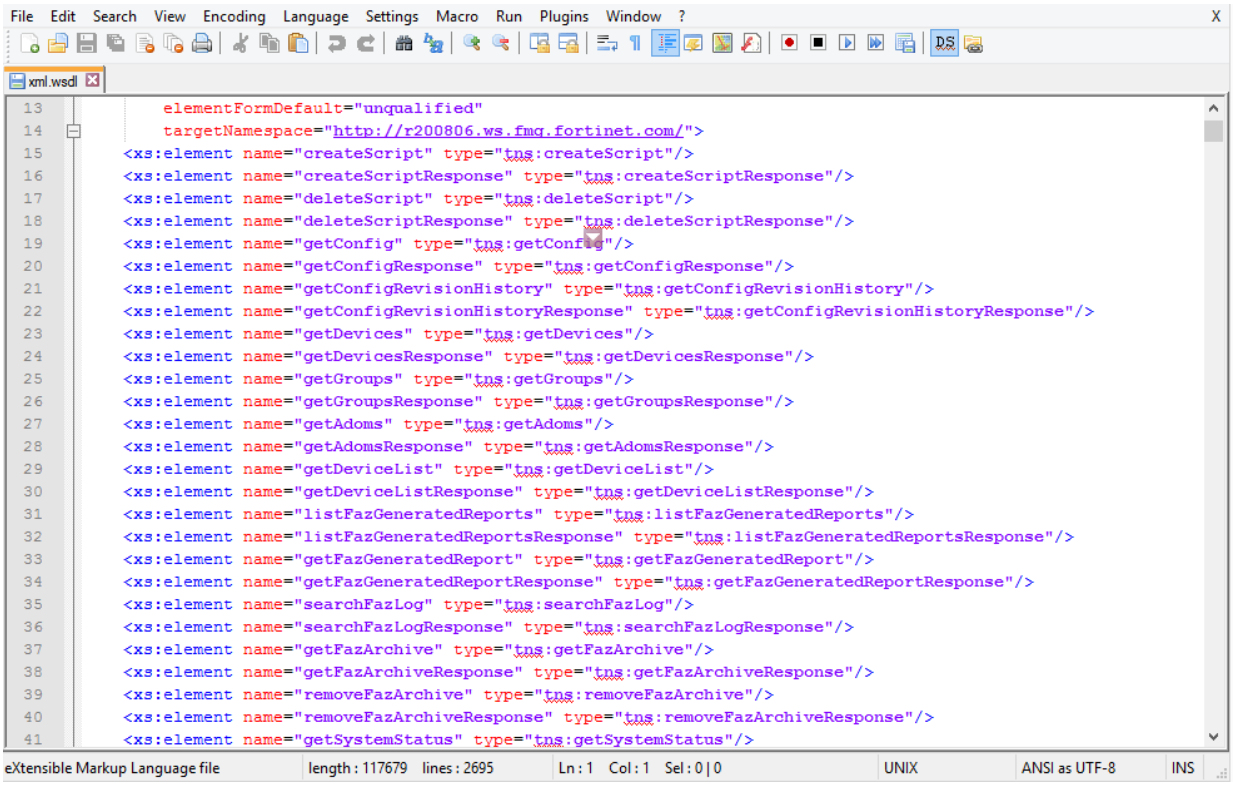
https://<FortiManager_ip_address>:8080/.

To download the WSDL file using the GUI:

1. Go to *System Settings* > *Advanced* > *Advanced Settings*
2. Select the *Download WSDL file* icon.



3. Save the `xml.wsdl` file to your local hard disk drive. You can open this file using a text editor.



By using a web testing tool, such as SoapUI, you can get information from your FortiManager.

Getting information from the FortiManager unit

To work with your managed devices, you need to obtain information from your FortiManager unit, such as:

- a list of ADOMs
- information about the managed devices
- information about individual devices
- the current configuration of devices, according to the database
- the revision history of devices.

SOAP error codes and descriptions

- SOAP_ERROR_OK = 0, /* same with SOAP_OK */
- SOAP_ERROR_DEFAUT_ZONE = -100, /* This is obsoleted */
- SOAP_ERROR_INVALID_PARAM = -101, /* invalid parameter(s) */
- SOAP_ERROR_PREPARE_PROBLEM = -102, /* prepare problem(s) */

- SOAP_ERROR_NOT_SUPPORTED = -103, /* not supported */
- SOAP_ERROR_FUNC_PROBLEM = -104, /* function problem */
- SOAP_ERROR_WRONG_CONDITION = -105, /* wrong condition(s) */
- SOAP_ERROR_MEMORY_LIMIT = -106, /* not enough memory */

Besides the *errorMsg* response, there could be errors returned in `<SOAP-ENV:Fault>` envelope as well. These are considered generic SOAP errors. There are also cases in which errors from the FortiManager application level are returned inside `<SOAP-ENV:Fault>` envelope. These errors are free-style; there are no error codes associated with them.

For example:

```
<SOAP-ENV:Fault>
   <faultcode>SOAP-ENV:Client</faultcode>
   <faultstring>Invalid admin uesr name '(null)'</faultstring>
   <detail>
   <error xmlns="http://localhost/">Invalid admin user name '(null)'</error>
   </detail>
</SOAP-ENV:Fault>
```

XML API elements

The following table lists the available FortiManager XML API elements.

XML API element	FortiManager v5.0.0	FortiManager v5.0.2 or later
addAdom	✓	✓
addDevice	✓	✓
addGroup	✓	✓
addPolicyPackage		✓
assignGlobalPolicy		✓
deleteAdom	✓	✓
deleteConfigRev	✓	✓
deleteDevice	✓	✓
deleteGroup	✓	✓
editAdom	✓	✓
editGroupMembership	✓	✓
getAdomList	✓	✓

XML API element	FortiManager v5.0.0	FortiManager v5.0.2 or later
getAdoms	✓	✓
getConfig	✓	✓
getConfigRevisionHistory	✓	✓
getDeviceLicenseList	✓	✓
getDeviceList	✓	✓
getDevices	✓	✓
getDeviceVdomList	✓	✓
getGroupList	✓	✓
getGroups	✓	✓
getInstLog	✓	✓
getPackageList	✓	✓
getSystemStatus		✓
getTaskList	✓	✓
getTCLRootFile	✓	✓
importPolicy	✓	✓
listRevisionId	✓	✓
retrieveConfig	✓	✓
revertConfig	✓	✓
getFazConfig		✓
setFazConfig		✓
runFazReport		✓
getFazGeneratedReports		✓
searchFazLog		✓
getFazArchive		✓

XML API element	FortiManager v5.0.0	FortiManager v5.0.2 or later
listFazGeneratedReports		✓
removeFazArchive		✓
createScript	✓	✓
deleteScript	✓	✓
getScript	✓	✓
getScriptLog	✓	✓
getScriptLogSummary	✓	✓
installConfig	✓	✓
runScript	✓	✓

FortiManager XML API Elements

addAdom

Use this request to add an ADOM to your FortiManager unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:addAdom>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <name>?</name>
         <version>?</version>
         <mr>?</mr>
         <!--Optional:-->
         <isBackupMode>?</isBackupMode>
         <!--Optional:-->
         <VPNManagement>?</VPNManagement>
         <!--Zero or more repetitions:-->
         <deviceSNVdom>
            <!--Optional:-->
            <SN>?</SN>
            <!--Zero or more repetitions:-->
            <vdomName>?</vdomName>
            <!--Zero or more repetitions:-->
```

```
            <vdomID>?</vdomID>
         </deviceSNVdom>
         <!--Zero or more repetitions:-->
         <deviceIDVdom>
            <!--Optional:-->
            <ID>?</ID>
            <!--Zero or more repetitions:-->
            <vdomName>?</vdomName>
            <!--Zero or more repetitions:-->
            <vdomID>?</vdomID>
         </deviceIDVdom>
      </r20:addAdom>
   </soapenv:Body>.
```

Request Field	Description
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<name>	The name of the ADOM to be created.
<version>	Firmware version options: • `400`: FortiOS version 4.0. • `500`: FortiOS version 5.0.
<mr>	The firmware major release version.
<isBackupMode>	Backup Mode ADOM options: • true: `BackupMode` is enabled. • false: `BackupMode` is disabled.
<VPNManagement>	VPN console ADOM options: • true: VPN console is enabled. • false: VPN console is disabled.
<deviceSNVdom>	XML structure consists of serial number, VDOM name, and VDOM ID variables.
<SN>	Serial number of device.
<vdomName>	The name of the VDOM.
<vdomID>	The VDOM identifier.
<deviceIDVdom>	XML structure consists of device ID, VDOM name, and `VDOM identifier` variables.
<ID>	The VDOM ID.

Request Field	Description
<vdomName>	The name of the VDOM.
<vdomID>	The VDOM identifier.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:addAdomResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Add members to adom test successfully</errorMsg>
         </errorMsg>
   </ns3:addAdomResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and details.
<errorCode>	Error code and message details: • 0: Added members to the ADOM successfully. • -101: The user does not have permission to run this command. Cannot get ADOM OID. • -102: The global workspace is locked. Cannot get ADOM detail information. • -104: Cannot get ADOM detail information. • -106: Not enough memory.
<errorMsg>	

addDevice

Use this request to add a device to your FortiManager unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:addDevice>
      <!--Optional:-->
         <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password></password>
      </servicePass>
      <!--Optional:-->
```

```
        <adom>root</adom>
        <!--Optional:-->
        <ip>1.1.1.1</ip>
        <!--Optional:-->
        <autod>manual</autod>
        <!--Optional:-->
        <deviceType>FortiGate</deviceType>
        <!--Optional:-->
        <name>test</name>
        <!--Optional:-->
        <adminUser>admin</adminUser>
        <!--Optional:-->
        <password></password>
        <!--Optional:-->
        <version>500</version>
        <!--Optional:-->
        <mr>0</mr>
        <!--Optional:-->
        <model>FortiGate-VM</model>
        <!--Optional:-->
        <flags></flags>
        <!--Optional:-->
        <description>sss</description>
        <!--Optional:-->
        <devId></devId>
        <!--Optional:-->
        <SN>FGVM001</SN>
        <!--Optional:-->
        <SNprefix>FGVM00</SNprefix>
     </r20:addDevice>
  </soapenv:Body>
```

Request Field	Description
<servicepass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<ip>	The device IP address.
<autod>	autod options: `true`, `false`, `manual`, or `unreg` Select if you want to enable auto discovery. The default value is `False`. Select the value `unreg` to promote an unregistered device.
<deviceType>	Select the type of device. The device type can be: FortiGate, FortiCarrier, or FortiSwitch.

Request Field	Description
<name>	The host name of the device.
<adminUser>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<version>	Firmware version options: • 400: FortiOS version 4.0. • 500: FortiOS version 5.0.
<mr>	The firmware major release version.
<model>	The device model number, FGT-60C, for example.
<flags>	Flags options: • harddisk: The device has a hard disk installed. • No value: Leave this field blank if the device does not have a hard disk installed.
<description>	The device description (optional).
<devId>	The device identifier.
<SN>	The device serial number.
<SNprefix>	The device serial number prefix.

The response is a series of <return> tags, each containing information about the device.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:addDeviceResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Read task ID 10 to get addDevice result</errorMsg>
         </errorMsg>
         <taskId>10</taskId>
   </ns3:addDeviceResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errormsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.

Response Field	Description
<errorCode>	Error code and message details: • 0: Read task ID to get add device result. • -101: The device IP cannot be empty. The device name must be input. Administrator user must be input. Unknown device type; only accepts `FortiGate`, `FortiCarrier`, or `FortiSwitch`. The device firmware version (400, or 500) must be input. The device version should be 400 or 500 value is invalid. The device version mr (0, 1, etc...) must be input. The device version major release is invalid. The device model (FortiGate-200B, FortiWiFi-60C, etc...) must be input. The device model is invalid. The device ID must be set when promoting an unregistered device. Promotable device does not exist. The device is not an unregistered device. • -102: The ADOM is locked. • -103: Add device auto discovery mode is not supported yet. • -104: Add device by IP in ADOM failed. Promote device by device ID in ADOM failed.
<errorMsg>	
<taskid>	Indicates the task ID number.

addGroup

Use this request to add a group to your FortiManager unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:addGroup>
         <servicePass>
            <userID>?</userID>
            <password>?</password>
         </servicePass>
            <adom>?</adom>
            <name>?</name>
            <description>?</description>
            <deviceSN>?</deviceSN>
            <deviceID>?</deviceID>
            <groupName>?</groupName>
            <groupID>?</groupID>
   </r20:addGroup>
</soapenv:Body>
```

Request Field	Description
<servicepass>	XML structure consists of username and password variables.

Request Field	Description
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<name>	The name of the group.
<description>	The group description.
<deviceSN>	The list of serial numbers of devices that belong to this group.
<deviceID>	The list of IDs of devices that belong to this group.
<groupName>	The list of names of sub-groups that belong to this group.
<groupID>	The list of IDs of sub-groups that belong to this group.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:addGroupResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Add group grp successfully</errorMsg>
         </errorMsg>
   </ns3:addGroupResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Added group name successfully. • -101: The group name cannot be empty. The input name is invalid. The name is in use. • -102: The ADOM is locked. • -104: Reached the number limit. The ADOM is locked. Add by name failed with error.
<errorMsg>	

addPolicyPackage

Use this request to add a policy package to your FortiManager unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:addPolicyPackage>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password>qa123456</password>
      </servicePass>
      <!--Optional:-->
      <adom>Adom50</adom>
      <!--Optional:-->
      <isGlobal>true</isGlobal>
      <!--Optional:-->
      <policyPackageName>Global_policy_package</policyPackageName>
      <!--Optional:-->
      <cloneFrom>default</cloneFrom>
      <!--Optional:-->
      <!-- <rename>?</rename> -->
      <!--Optional:-->
      <packageInstallTarget>
         <!--Zero or more repetitions:-->
         <!-- <grp> -->
            <!--Optional:-->
            <!-- <oid>?</oid> -->
            <!--Optional:-->
            <!-- <name>?</name> -->
         <!-- </grp> -->
         <!--Zero or more repetitions:-->
         <dev>
            <!--Optional:-->
            <!-- <oid>?</oid> -->
            <!--Optional:-->
            <name>FortiGate-VM-70</name>
            <!--Zero or more repetitions:-->
            <!-- <vdom> -->
               <!--Optional:-->
               <!-- <oid>?</oid> -->
               <!--Optional:-->
               <!-- <name>?</name> -->
            <!-- </vdom> -->
         </dev>
      </packageInstallTarget>
   </r20:addPolicyPackage>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<isGlobal>	Add a global policy package.
<policyPackageName>	The policy package name.
<cloneFrom>	Clone from policy name.
<rename>	Rename the policy package.
<packageInstallTarget>	Package install target includes group and VDOM targets.
<grp>	
<oid>	The object identifier.
<name>	The group name.
<vdom>	
<oid>	The object identifier.
<name>	The name of the VDOM.

assignGlobalPolicy

Use this request to assign a global policy package to your FortiManager unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:assignGlobalPolicy>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password>Password</password>
```

```
        </servicePass>
        <!--Optional:-->
        <adom>Adom5x2</adom>
        <!--Optional:-->
        <!-- <policyPackageName>?</policyPackageName> -->
        <!--Optional:-->
        <!-- <policyPackageOid>?</policyPackageOid> -->
        <!--Zero or more repetitions:-->
        <!-- <adomList> -->
           <!--Optional:-->
           <!-- <oid>?</oid> -->
           <!--Optional:-->
           <!-- <name>?</name> -->
        <!-- </adomList> -->
        <!--Optional:-->
        <!-- <allObjects>?</allObjects> -->
        <!--Optional:-->
        <!-- <installToDevice>?</installToDevice> -->
        <!--Optional:-->
        <!-- <checkAssigndDup>?</checkAssigndDup> -->
     </r20:assignGlobalPolicy>
  </soapenv:Body>
```

Request Field	Description
<servicepass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<policyPackageName>	The policy package name.
<policyPackageOid>	The policy package object identifier.
<adomList>	
<oid>	The object identifier.
<name>	The ADOM name.
<allObjects>	
<installToDevice>	Install policy to device.
<checkAssigndDup>	

deleteAdom

Use this request to delete an ADOM from your FortiManager unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:deleteAdom>
         <servicePass>
            <userID>?</userID>
            <password>?</password>
         </servicePass>
            <adomName>?</adomName>
            <adomOid>?</adomOid>
   </r20:deleteAdom>
</soapenv:Body>
```

Request Field	Description
<servicepass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adomName>	The name of the ADOM.
<adomOid>	The ADOM object identifier (OID).

The response indicates with the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:deleteAdomResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Delete adom ID 166 successfully</errorMsg>
         </errorMsg>
   </ns3:deleteAdomResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.

Response Field	Description
<errorCode>	Error code and message details: • 0: Deleted ADOM ID successfully. • -101: The ADOM name is invalid. Cannot get a valid ADOM ID. Invalid ADOM. • -102: The ADOM is locked. The global workspace is locked. • -104: The ADOM ID cannot be deleted. • -105: The root ADOM cannot be deleted. The ADOM ID is in use and cannot be deleted.
<errorMsg>	

deleteConfigRev

Use this request to delete a configuration revision defined on your FortiManager unit. Only an administrator with the `Super_User` profile can run this command.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:deleteConfigRev>
         <servicePass>
            <password>?</password>
            <userID>?</userID>
         </servicePass>
            <devId>?</devId>
            <serialNumber>?</serialNumber>
            <revName>?</revName>
            <revId>?</revId>
   </r20:deleteConfigRev>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<devId>	The device ID.
<serialNumber>	Serial number of device.
<revName>	The revision name. You can get this in the *Revision History* section in the GUI.
<revId>	The revision ID.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:deleteConfigRevResponse>
      <errorMsg>
      <errorCode>0</errorCode>
      <errorMsg>Delete revision 1 from device ID 129 successfully</errorMsg>
      </errorMsg>
   </ns3:deleteConfigRevResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Deleted revision ID from device ID successfully • -101: Need a valid revision name or revision ID. • -102: The ADOM is locked. The device is in backup mode. No revision ID matched with revision name. No valid revision ID exists. The revision ID does not exist on device ID. • -104: User deleted revision ID on device ID failed.
<errorMsg>	

deleteDevice

Use this request to delete a device defined on your FortiManager unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:deleteDevice>
         <servicePass>
            <password></password>
            <userID>admin</userID>
         </servicePass>
            <devId>468985</devId>
            <serialNumber>FGT60C3G06500185</serialNumber>
   </r20:deleteDevice>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.

Request Field	Description
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<devId>	The device ID number.
<serialNumber>	The serial number of device.

The response indicates if the device was deleted successfully or if the procedure failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:deleteDeviceResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Delete device ID 468985 successfully</errorMsg>
         </errorMsg>
   </ns3:deleteDeviceResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Read task ID to get delete device result. • -102: The ADOM is locked. • -104: The device can only be deleted from the ADOM which contains its root VDOM. The device ID cannot be deleted. • -105: The device ID is in use and cannot be deleted. The device ID was locked and cannot be deleted.
<errorMsg>	

deleteGroup

Use this request to delete a group from your FortiManager unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:deleteGroup>
         <servicePass>
            <userID>?</userID>
            <password>?</password>
         </servicePass>
            <adom>?</adom>
            <name>?</name>
```

```
            <grpId>?</grpId>
      </r20:deleteGroup>
   </soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<name>	The name of the group to be deleted.
<grpId>	The group identifier.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:deleteGroupResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Delete group ID 179 successfully</errorMsg>
         </errorMsg>
   </ns3:deleteGroupResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Deleted group ID successfully. • -101: The group name is invalid. Cannot locate a valid group from group ID and name information. • -102: The ADOM is locked. • -104: The group ID cannot be deleted. • -105: The group ID is in use and cannot be deleted. The group ID was locked and cannot be deleted.
<errorMsg>	

editAdom

Use this request to edit an ADOM.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:editAdom>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <name>?</name>
         <!--Optional:-->
         <version>?</version>
         <!--Optional:-->
         <mr>?</mr>
         <!--Optional:-->
         <state>?</state>
         <!--Optional:-->
         <isBackupMode>?</isBackupMode>
         <!--Optional:-->
         <VPNManagement>?</VPNManagement>
         <!--Optional:-->
         <metafields>
            <!--Zero or more repetitions:-->
            <metafield>
               <name>?</name>
               <value>?</value>
            </metafield>
         </metafields>
         <!--Zero or more repetitions:-->
         <addDeviceSNVdom>
            <!--Optional:-->
            <SN>?</SN>
            <!--Zero or more repetitions:-->
            <vdomName>?</vdomName>
            <!--Zero or more repetitions:-->
            <vdomID>?</vdomID>
            </addDeviceSNVdom>
         <!--Zero or more repetitions:-->
         <addDeviceIDVdom>
            <!--Optional:-->
            <ID>?</ID>
            <!--Zero or more repetitions:-->
            <vdomName>?</vdomName>
            <!--Zero or more repetitions:-->
            <vdomID>?</vdomID>
         </addDeviceIDVdom>
   </r20:editAdom>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password • Leave field blank for no password
<name>	The name of the ADOM to be edited.
<version>	Firmware version options: • `400`: FortiOS version 4.0. • `500`: FortiOS version 5.0.
<mr>	The firmware major release version.
<state>	Device ADOM state options: • true: ADOMs are enabled • false: ADOMs are disabled
<isBackupMode>	Backup Mode ADOM options: • true: `BackupMode` is enabled. • false: `BackupMode` is disabled.
<VPNManagement>	VPN console ADOM options: • true: VPN console is enabled. • false: VPN console is disabled.
<metafields>	XML structure consists of metafield data. These strings occur in pairs in XML responses.
<name>	Name of device metafield (s).
<value>	Value of device metafield (s).
<addDeviceSNVdom>	XML structure consists of serial number, VDOM name, and VDOM ID variables.
<SN>	Serial number of device, FGT60C3G06500185, for example.
<vdomName>	The name of the VDOM.
<vdomID>	The VDOM identifier.
<addDeviceIDVdom>	XML structure consists of the device ID, VDOM name, and VDOM ID variables.

Request Field	Description
<ID>	The ID of the device.
<vdomName>	The name of the VDOM.
<vdomID>	The VDOM identifier.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:editAdomResponse>
         <errorMsg>
         <errorCode>0</errorCode>
            <errorMsg>Edit adom root successfully</errorMsg>
         </errorMsg>
   </ns3:editAdomResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode> <errorMsg>	Error code and message details: • 0: Edited ADOM name successfully. • -101: The ADOM name cannot be empty. The ADOM name is invalid. Version only accepts 400 or 500 values. Invalid major release value. The ADOM metafield does not exist. The metafield name does not exist. Failed to change ADOM information. • -102: The global workspace is locked. Failed to get ADOM information. Failed to get ADOM flags. Failed to create device fetch. Cannot change mode to backup mode since the ADOM has device(s). Cannot get ADOM metafields. • -104: Adding members to ADOM failed.

editGroupMembership

Use this request to edit group membership.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:editGroupMembership>
         <servicePass>
            <userID>?</userID>
            <password>?</password>
```

```
            </servicePass>
               <adom>?</adom>
               <name>?</name>
               <grpId>?</grpId>
               <addDeviceSNList>?</addDeviceSNList>
               <addDeviceIDList>?</addDeviceIDList>
               <delDeviceSNList>?</delDeviceSNList>
               <delDeviceIDList>?</delDeviceIDList>
               <addGroupNameList>?</addGroupNameList>
               <addGroupIDList>?</addGroupIDList>
               <delGroupNameList>?</delGroupNameList>
               <delGroupIDList>?</delGroupIDList>
         </r20:editGroupMembership>
      </soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password • Leave field blank for no password
<adom>	The name of the ADOM, or if the ADOM status is disabled, then enter `root`. If the ADOM field is empty or does not match an ADOM name, the response lists all devices.
<name>	The name of the group to be edited.
<grpID>	The group identifier.
<addDeviceSNList>	The device serial number list to add.
<addDeviceIDList>	The device identifier list to add.
<delDeviceSNList>	The device serial number list to delete.
<delDeviceIDList>	The device identifier list to delete.
<addGroupNameList>	The group name list to add.
<addGroupIDList>	The group identifier list to add.
<delGroupNameList>	The group name list to delete.
<delGroupIDList>	The group identifier list to delete.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
```

```
<SOAP-ENV:Body>
   <ns3:editGroupMembershipResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Edit group grp successfully</errorMsg>
         </errorMsg>
   </ns3:editGroupMembershipResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Edited group name successfully. • -101: Cannot find the group by the provided name or ID. • -102: The ADOM is locked.
<errorMsg>	

getAdomList

Use this request to get a list of the ADOMs defined on your FortiManager unit. Only an administrator with the `Super_User` profile can run this command.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getAdomList>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <detail>?</detail>
   </r20:getAdomList>
</soapenv:Body>
```

Request Field	Description
<servicePass	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password • Leave field blank for no password
<detail>	Detail field options: true or false

The response is a series of <return> tags, each containing information about an ADOM.

Example response: detail is true

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getAdomListResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>get adom detail list successfully</errorMsg>
         </errorMsg>
         <adomDetail>
            <oid>103</oid>
            <name>others</name>
            <description/>
            <version>500</version>
            <mr>0</mr>
            <state>true</state>
            <isBackupMode>false</isBackupMode>
            <VPNManagement>true</VPNManagement>
            <metafields>
               <metafield>
                  <name>meta1</name>
                  <value/>
               </metafield>
            </metafields>
         </adomDetail>
         <adomDetail>
            <oid>3</oid>
            <name>root</name>
            <description/>
            <version>500</version>
            <mr>0</mr>
            <state>true</state>
            <isBackupMode>false</isBackupMode>
            <VPNManagement>true</VPNManagement>
            <metafields>
            <metafield>
               <name>meta1</name>
               <value/>
            </metafield>
         </metafields>
         </adomDetail>
         <adomDetail>
            <oid>160</oid>
            <name>test1</name>
            <description/>
            <version>500</version>
            <mr>1</mr>
            <state>true</state>
            <isBackupMode>false</isBackupMode>
            <VPNManagement>false</VPNManagement>
            <metafields>
               <metafield>
                  <name>meta1</name>
                  <value>me</value>
               </metafield>
```

```
            </metafields>
            </adomDetail>
      </ns3:getAdomListResponse>
   </SOAP-ENV:Body>
```

Example response: (detail is false)

```
   <SOAP-ENV:Header/>
   <SOAP-ENV:Body>
      <ns3:getAdomListResponse>
            <errorMsg>
               <errorCode>0</errorCode>
               <errorMsg>get adom list successfully</errorMsg>
            </errorMsg>
            <adomInfo>
               <oid>103</oid>
               <name>others</name>
               <description/>
               <version>500</version>
               <mr>0</mr>
               <state>true</state>
            </adomInfo>
            <adomInfo>
               <oid>3</oid>
               <name>root</name>
               <description/>
               <version>500</version>
               <mr>0</mr>
               <state>true</state>
            </adomInfo>
            <adomInfo>
               <oid>160</oid>
               <name>test1</name>
               <description/>
               <version>500</version>
               <mr>1</mr>
               <state>true</state>
            </adomInfo>
      </ns3:getAdomListResponse>
   </SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Retrieved ADOM list successfully. Retrieved ADOM detail list successfully. • -104: ADOM fetch error. Cannot get ADOM basic information. Cannot get ADOM detail information. • -106: Not enough memory.
<errorMsg>	

Response Field	Description
<adomDetail>	XML structure consists of the object identifier, ADOM name, and description.
<oid>	The object identifier.
<name>	The ADOM name.
<description>	A description of the ADOM.
<version>	Firmware version options: • `400`: FortiOS version 4.0. • `500`: FortiOS version 5.0.
<mr>	The firmware major release version.
<state>	Device ADOM state options: • true: ADOMs are enabled. • false: ADOMs are disabled.
<isBackupMode>	Backup Mode ADOM options: • true: `BackupMode` is enabled. • false: `BackupMode` is disabled.
<VPNManagement>	VPN console ADOM options: • true: VPN console is enabled. • false: VPN console is disabled.
<metafield>	XML structure consists of metafield data. These strings occur in pairs in XML responses.
<name>	Name of device metafield (s).
<value>	Value of device metafield (s).

getAdoms

Use this request to get a list of ADOMs.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getAdoms>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
```

```
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Zero or more repetitions:-->
         <names>?</names>
         <!--Zero or more repetitions:-->
         <adomIds>?</adomIds>
      </r20:getAdoms>
   </soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<names>	The ADOM name.
<adomIDs>	The ADOM object ID.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getAdomsResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Get adoms info Successfully</errorMsg>
         </errorMsg>
         <adomDetail>
            <oid>3</oid>
            <name>root</name>
            <description/>
            <version>5</version>
            <mr>0</mr>
            <state>true</state>
            <isBackupMode>false</isBackupMode>
            <VPNManagement>true</VPNManagement>
         <metafields>
            <metafield>
               <name>meta1</name>
               <value/>
            </metafield>
         </metafields>
         </adomDetail>
   </ns3:getAdomsResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Retrieved ADOM information successfully. • -101: Invalid admin user name. User does not have permission to run this command. Cannot get ADOM OID. • -102: Cannot get ADOM detail information. • -104: Cannot get ADOM detail information. • -106: Not enough memory.
<errorMsg>	
<adomDetail>	XML structure consists of the object identifier, ADOM name, description, firmware version, and major release.
<oid>	The object identifier for the ADOM.
<name>	The name of the ADOM.
<description>	A description of the ADOM.
<version>	Firmware version options: • `400`: FortiOS version 4.0. • `500`: FortiOS version 5.0.
<mr>	The firmware major release version.
<state>	Device ADOM state options: • true: ADOMs are enabled. • false: ADOMs are disabled.
<isBackupMode>	Backup Mode ADOM options: • true: `BackupMode` is enabled. • false: `BackupMode` is disabled.
<VPNManagement>	VPN console ADOM options: • true: VPN console is enabled. • false: VPN console is disabled.
<metafield>	XML structure consists of metafield data. These strings occur in pairs in XML responses.
<name>	Name of device metafield (s).
<value>	Value of device metafield (s).

getConfig

Use this request to retrieve a particular revision of the device configuration from the device database.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getConfig>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <devId>?</devId>
         <!--Optional:-->
         <serialNumber>?</serialNumber>
         <!--Optional:-->
         <adom>?</adom>
         <!--Optional:-->
         <revisionNumber>?</revisionNumber>
   </r20:getConfig>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<devId>	The Device ID. This is the primary device identifier. You can omit this field and use the serial number instead.
<serialNumber>	Serial number of device, FGT60C3G06500185, for example. This device identifier is secondary to device.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<revisionNumber>	The revision that you want to view. Use a negative number to retrieve the latest revision.

The response is a <return> field containing the device configuration and other information.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getConfigResponse>
         <return>
            <branchPoint>128</branchPoint>
            <checkinDate>2012-11-02T19:37:39Z</checkinDate>
            <checkinUser>admin</checkinUser>
            <content>
            ... configuration file content ...
            </content>
            <message/>
         <oid>109</oid>
         <osVersion>5</osVersion>
         <platform>FortiGate-60C</platform>
         <revisionNum>3</revisionNum>
         <serialNumber>FGT60C3G06500185</serialNumber>
         </return>
   </ns3:getConfigResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<branchPoint>	The firmware build number, except for some special branch builds.
<checkinDate>	Date and time (UTC) when this revision was installed to the device.
<checkinUser>	The userID of the administrator who installed this revision.
<content>	The device configuration file contents.
<oid>	The object identifier.
<osVersion>	Version of device operating system, 5, for example for FortiOS 5.0.
<platform>	Platform name for device, FortiGate-60C, for example.
<revisionNum>	Configuration revision ID.
<serialNumber>	Serial number of device, FGT60C3G06500185, for example.

getConfigRevisionHistory

Use this request to retrieve multiple revisions of the device configuration from the device database. You can retrieve based on revision numbers or check-in times.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getConfigRevisionHistory>
         <servicePass>
```

```
            <password></password>
            <userID>admin</userID>
         </servicePass>
         <serialNumber>FGT60C3G06500185</serialNumber>
         <!-- <devId>?</devId> -->
         <!-- <checkinUser></checkinUser> -->
         <!-- <minCheckinDate></minCheckinDate> -->
         <!-- <maxCheckinDate></maxCheckinDate> -->
         <!-- <minRevisionNumber>1</minRevisionNumber> -->
         <!-- <maxRevisionNumber>?</maxRevisionNumber> -->
      </r20:getConfigRevisionHistory>
   </soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<serialNumber>	Serial number of device, FGT60C3G06500185, for example. This device identifier is secondary to device.
<devId>	The Device ID. This is the primary device identifier.
<checkinUser>	Optionally, specify the user ID of the administrator who saved this revision.
<minCheckinDate>	Optionally, specify the earliest revision check-in time to retrieve. Use with `<maxCheckinDate>`.
<maxCheckinDate>	Optionally, specify the latest revision check-in time to retrieve. Use with `<minCheckinDate>`.
<minRevisionNumber>	Optionally, specify the first revision to retrieve. Use with `<maxRevisionNumber>`.
<maxRevisionNumber>	Optionally, specify the last revision to retrieve. Use with `<minRevisionNumber>`.

The response is a series of <return> fields containing the device configuration and other information.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getConfigRevisionHistoryResponse>
         <return>
            <branchPoint>128</branchPoint>
            <checkinDate>2012-11-02T19:37:39Z</checkinDate>
            <checkinUser>admin</checkinUser>
            <content>
```

```
            ... configuration file content - newest retrieved revision ...
            </content>
            <message/>
         <oid>109</oid>
         <osVersion>5</osVersion>
         <platform>FortiGate-60C</platform>
         <revisionNum>3</revisionNum>
         <serialNumber>FGT60C3G06500185</serialNumber>
         </return>
         <return>
            ... information about preceding revision ...
</return>
```

Response Field	Description
<branchPoint>	The firmware build number, except for some special branch builds.
<checkinDate>	Date and time (UTC) when this revision was installed to the device.
<checkinUser>	The userID of the administrator who installed this revision.
<content>	The device configuration file contents.
<oid>	The object identifier.
<osVersion>	Version of device operating system, 5, for example for FortiOS 5.0.
<platform>	Platform name for device, FortiGate-60C, for example.
<revisionNum>	Configuration revision ID.
<serialNumber>	Serial number of device, FGT60C3G06500185, for example.
<return>	Information about the preceding revision.

getDeviceLicenseList

Use this request to obtain a list of device licenses.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getDeviceLicenseList>
         <servicePass>
            <userID>?</userID>
            <password>?</password>
   </r20:getDeviceLicenseList>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.

The response includes the device serial number, support type, support level, and expiry date.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getDeviceLicenseListResponse>
      <return>
         <device>
            <serial_number>FGT60C3G06500185</serial_number>
            <contract>
               <support_type>AVDB</support_type>
               <support_level>99</support_level>
               <expiry_date>20300103</expiry_date>
            </contract>
            <contract>
               <support_type>AVEN</support_type>
               <support_level>99</support_level>
               <expiry_date>20300103</expiry_date>
            </contract>
            <contract>
               <support_type>COMP</support_type>
               <support_level>99</support_level>
               <expiry_date>20300903</expiry_date>
            </contract>
            <contract>
               <support_type>ENHN</support_type>
               <support_level>99</support_level>
               <expiry_date>20300803</expiry_date>
            </contract>
            <contract>
               <support_type>FMWR</support_type>
               <support_level>99</support_level>
               <expiry_date>20300403</expiry_date>
            </contract>
            <contract>
               <support_type>FRVS</support_type>
               <support_level>99</support_level>
               <expiry_date>20300503</expiry_date>
            </contract>
            <contract>
               <support_type>FURL</support_type>
               <support_level>99</support_level>
               <expiry_date>20301103</expiry_date>
            </contract>
```

```
            <contract>
               <support_type>HDWR</support_type>
               <support_level>99</support_level>
               <expiry_date>20301203</expiry_date>
            </contract>
            <contract>
               <support_type>NIDS</support_type>
               <support_level>99</support_level>
               <expiry_date>20300303</expiry_date>
            </contract>
            <contract>
               <support_type>SPAM</support_type>
               <support_level>99</support_level>
               <expiry_date>20300203</expiry_date>
            </contract>
            <contract>
               <support_type>SPRT</support_type>
               <support_level>99</support_level>
               <expiry_date>20300903</expiry_date>
            </contract>
            <contract>
               <support_type>VCME</support_type>
               <support_level>99</support_level>
               <expiry_date>20301003</expiry_date>
            </contract>
         </device>
      </return>
   </ns3:getDeviceLicenseListResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<serial_number>	The device serial number.
<support_type>	Support contract types include: • AVDB: Antivirus Signature Definition Update Support • AVEN: Antivirus Engine Update Support • COMP: Comprehensive Support • ENHN: Enhancement Support • FMWR: Firmware Update Support • FRVS: FortiScanner Database Update Support • FURL: Web Filtering Support • SPAM: AntiSpam Support • HDWR: Hardware Support • NIDS: Intrusion Detection Support • SPRT: Technical Support via Telephone • VCME: FortiGate Network scanner plugin

Response Field	Description
<support_level>	Support levels include: • 99: Trial contract • 10: 8x5 support contract • 20: 24x7 support contract
<expiry_date>	Support contract expiry date.

getDeviceList

Use this request to get summary information about the managed devices, optionally limited to a particular ADOM.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getDeviceList>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <adom>?</adom>
         <!--Optional:-->
         <detail>?</detail>
   </r20:getDeviceList>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<detail>	Detail field options: true, or false

The response is a series of <return> tags, each containing information about a device.

Example response: (detail is true)

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getDeviceListResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>get device detail list successfully</errorMsg>
         </errorMsg>
         <deviceDetail>
            <devId>129</devId>
            <firmware>FortiGate</firmware>
            <firmwareVersion>5</firmwareVersion>
            <buildNum>128</buildNum>
            <description/>
            <hostname>FGT60C3G06500185</hostname>
            <platform>FortiGate-60C</platform>
            <sn>FGT60C3G06500185</sn>
            <ip>10.2.60.99</ip>
            <IPSContract>3.00249(2012-10-11 02:47)</IPSContract>
            <antiVirusContract>16.00560(2012-10-19 08:31)</antiVirusContract>
            <appsignature/>
            <mgmtMode>reg</mgmtMode>
         </deviceDetail>
   </ns3:getDeviceListResponse>
</SOAP-ENV:Body>
```

Example response: (detail is false)

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getDeviceListResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>get device list successfully</errorMsg>
         </errorMsg>
         <deviceInfo>
            <devId>129</devId>
            <firmware>FortiGate</firmware>
            <firmwareVersion>5</firmwareVersion>
            <buildNum>128</buildNum>
            <description/>
            <hostname>FGT60C3G06500185</hostname>
            <platform>FortiGate-60C</platform>
            <sn>FGT60C3G06500185</sn>
            <ip>10.2.60.99</ip>
         </deviceInfo>
   </ns3:getDeviceListResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.

Response Field	Description
<errorCode>	Error code and message details: • 0: Retrieved device list successfully. Retrieved device detail list successfully. • -102: Device fetch error for ADOM. • -104: Cannot get device detail information.
<errorMsg>	
<deviceDetail>	XML structure consists of the following tags.
<devID>	The Device ID. This is the primary device identifier.
<firmware>	FortiGate, FortiCarrier, or FortiSwitch
<firmwareVersion>	Version of device operating system, 5, for example for FortiOS 5.0.
<buildNum>	Firmware version build number, 0128, for example.
<description>	Device description from FortiManager database.
<hostname>	The device host name.
<platform>	Platform name for device, FortiGate-60C, for example.
<sn>	Serial number of device, FGT60C3G06500185, for example.
<ip>	IP address of device network interface from which response was received.
<IPSContract>	FortiGuard IPS definitions version and last update time, 2.00461(2008-12-08 11:23), for example.
<antiVirusContract>	AV contract and expiry date, 8.00631(2012-02-15 14:27), for example.
<appsignature>	FortiGuard application signature.
<mgmtMode>	The device management mode. One of the following: • reg: Registered device • unreg: Unregistered device • unknown: Device registration status is unknown.

getDevices

Use this request to get information about specific managed devices, identified by serial number or device ID. You can obtain device ID values by using the `execute dmserver showdev` CLI command.

If you want information about the device's configuration, see getDevices on page 46.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getDevices>
         <servicePass>
            <password></password>
            <userID>admin</userID>
         </servicePass>
         <serialNumbers>FGT60C3G06500185</serialNumbers>
         <serialNumbers>FGT60C3G06500186</serialNumbers>
   </r20:getDevices>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<serialNumbers>	Serial number of the device. This is the secondary identifier. You can enter multiple serial numbers fields.
<devIds>	Device ID. This is the primary device identifier. You can omit this field and use the serial number instead. You can enter multiple device ID fields.

The response is a series of <return> tags, each containing information about a device.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getDevicesResponse>
         <return>
            <firmware>FortiGate</firmware>
            <firmwareVersion>5</firmwareVersion>
            <buildNum>128</buildNum>
            <description/>
            <hostname>Dev3</hostname>
            <IPSContract>2.442(2012-11-08 11:23)</IPSContract>
            <antiVirusContract>8.368(2007-11-15 13:59)</antiVirusContract>
            <platform>FortiGate-60C</platform>
            <sn>FGT60C3G06500185</sn>
            <ip>172.20.120.126</ip>
         </return>
         <return>
            <firmware>FortiGate</firmware>
            <firmwareVersion>5</firmwareVersion>
            <buildNum>128</buildNum>
            <description/>
            <hostname>FGT60C3G06500185</hostname>
            <IPSContract/>
```

```
            <antiVirusContract/>
            <platform>Fortigate-60C</platform>
            <sn>FGT60C3G06500186</sn>
            <ip>172.20.120.127</ip>
         </return>
      </ns3:getDevicesResponse>
   </SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Retrieved device(s) information successfully. • -102: Serial number is not found. Cannot get device information. • -104: Cannot get device information.
<errorMsg>	
<firmware>	One of: • FortiGate • FortiCarrier • FortiSwitch
<firmwareVersion>	Version of device operating system, 500, for example for FortiOS 5.0.
<buildNum>	Firmware version build number, 0128, for example.
<description>	Device description from database.
<hostname>	The device host name.
<IPSContract>	FortiGuard IPS definitions version and last update time, 2.00461(2012-11-08 11:23), for example.
<antiVirusContract>	AV contract and expiry date, 8.00631(2012-02-15 14:27), for example.
<platform>	Platform name for device, FortiGate-60C, for example.
<sn>	Serial number of device, FGT60C3G06500185, for example.
<ip>	IP address of device network interface from which response was received.

getDeviceVdomList

Use this request to obtain a list of device VDOMs.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getDeviceVdomList>
```

```
            <!--Optional:-->
            <servicePass>
               <!--Optional:-->
               <userID>?</userID>
               <!--Optional:-->
               <password>?</password>
            </servicePass>
            <!--Optional:-->
            <devName>?</devName>
            <!--Optional:-->
            <devID>?</devID>
      </r20:getDeviceVdomList>
   </soapenv:Body>
```

Request Field	Description
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<user>	The administrator user name.
<devName>	Name of the device host.
<devId>	The Device ID. This is the primary device identifier.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getDeviceVdomListResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Get device 114 vdom list successfully</errorMsg>
         </errorMsg>
         <name>FGT60C3G06500185</name>
         <oid>114</oid>
         <return>
            <name>root</name>
            <oid>3</oid>
         </return>
   </ns3:getDeviceVdomListResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.

Response Field	Description
<errorCode>	Error code and message details: • 0: Retrieved device VDOM list successfully. • -101: Cannot find the device by provided name or ID.
<errorMsg>	
<name>	The name of the VDOM device list.
<oid>	The object identifier.

getGroupList

Use this request to obtain a list of device groups, optionally limited to a particular ADOM.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getGroupList>
      <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
      <!--Optional:-->
         <adom>?</adom>
         <!--Optional:-->
         <detail>?</detail>
   </r20:getGroupList>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<detail>	Detail field options: true or false.

The response is a series of <return> fields, each listing one group, in ascending order of object identifier (OID). Both built-in groups, like All FortiGate, and user-defined groups are listed.

Example response: (detail is true)

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getGroupListResponse>
      <errorMsg>
         <errorCode>0</errorCode>
         <errorMsg>get group detail list successfully</errorMsg>
      </errorMsg>
      <groupDetail>
         <oid>102</oid>
         <name>All_FortiCarrier</name>
      </groupDetail>
      <groupDetail>
         <oid>101</oid>
         <name>All_FortiGate</name>
      <devMemberList>
         <oid>129</oid>
         <name>FGT60C3G06500185</name>
      </devMemberList>
      </groupDetail>
   </ns3:getGroupListResponse>
</SOAP-ENV:Body>
```

Example response: (detail is false)

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getGroupListResponse>
      <errorMsg>
         <errorCode>0</errorCode>
         <errorMsg>get group list successfully</errorMsg>
      </errorMsg>
      <groupInfo>
         <oid>102</oid>
         <name>All_FortiCarrier</name>
      </groupInfo>
      <groupInfo>
         <oid>101</oid>
         <name>All_FortiGate</name>
      </groupInfo>
   </ns3:getGroupListResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.

Response Field	Description
<errorCode>	Error code and message details: • 0: Retrieved group list successfully. Retrieved group detail list successfully. • -102: Group fetch error for ADOM. • -105: Failed to get group detail information. • -106: Not enough memory.
<errorMsg>	
<groupDetail>	XML structure consists of the object identifier, and name.
<oid>	The object identifier.
<name>	The group name.
<devMemberList>	XML structure consists of the object identifier, and name.
<oid>	The object identifier.
<name>	The device member list name.

getGroups

Use this request to obtain a list a groups.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getGroups>
      <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>admin</userID>
            <!--Optional:-->
            <password></password>
         </servicePass>
         <!--Optional:-->
         <adom>root</adom>
         <!--Zero or more repetitions:-->
         <names>grp</names>
         <!--Zero or more repetitions:-->
      <grpIds>237</grpIds>
   </r20:getGroups>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.

Request Field	Description
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then ADOM is root.
<names>	The names of a list of groups.
<grpIds>	The group IDs.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getGroupsResponse>
      <errorMsg>
         <errorCode>0</errorCode>
         <errorMsg>get group(s) details sucessfully</errorMsg>
      </errorMsg>
      <groupDetail>
         <oid>237</oid>
         <name>grp</name>
      <devMemberList>
         <oid>206</oid>
         <name>FGT60C3G06500185</name>
      </devMemberList>
      </groupDetail>
   </ns3:getGroupsResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Retrieved group(s) details successfully. • -102: Cannot get group information for group ID. • -104: Failed to get group detail information. • -106: Not enough memory.
<errorMsg>	
<groupDetail>	XML structure consists of the object identifier, and name.
<oid>	The object identifier.

Response Field	Description
<name>	The group name.
<devMemberList>	XML structure consists of the object identifier, and name.
<oid>	The object identifier.
<name>	The device member list name.

getInstLog

Use this request to obtain the install log.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getInstlog>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password></password>
      </servicePass>
      <!--Optional:-->
      <devId>2992</devId>
      <!--Optional:-->
      <serialNumber>FGT60C3G06500185</serialNumber>
      <!--Optional:-->
      <taskId>286</taskId>
   </r20:getInstlog>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<devID>	The device ID. This is the primary device identifier.
<serialNumber>	Serial number of device, FGT60C3G06500185, for example.
<taskID>	Indicates the task ID number.

The response contains details of the installation.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getInstlogResponse>
      <instLog>
         <content>Starting log (Run on device)

   Start installing
   FGT60C3G06500185 $ config system global
   FGT60C3G06500186 (global) $ set admintimeout 80
   FGT60C3G06500187 (global) $ end

   ---> generating verification report
   &lt;--- done generating verification report

   install finished</content>
      </instLog>
   </ns3:getInstlogResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<instLog>	XML structure consists of a content tag with information on the installation log.
<content>	Details of the installation.

getPackageList

Use this request to retrieve a package list.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getPackageList>
         <servicePass>
            <userID>?</userID>
            <password>?</password>
         </servicePass>
         <adom>?</adom>
         <isGlobal>?</isGlobal>
   </r20:getPackageList>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.

Request Field	Description
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<adom>	If the ADOM field is blank, it is assigned as root.
<isGlobal>	Set for global policy package list. Enter either true or false.

The response includes the object identifier, package list name, and package type.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getPackageListResponse>
      <return>
         <oid>521</oid>
         <name>default</name>
         <type>package</type>
      </return>
      <return>
         <oid>568</oid>
         <name>FGT60C3G06500185_root_0</name>
         <type>package</type>
      </return>
      <return>
         <oid>574</oid>
         <name>FGT60C3G06500186_root_1</name>
         <type>package</type>
      </return>
   </ns3:getPackageListResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<return>	XML structure consists of the object identifier, name, and type.
<oid>	The object identifier.
<name>	The package list name.
<type>	The package type.

getSystemStatus

Use this request to get system status information in the current system.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getSystemStatus>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>admin</userID>
            <!--Optional:-->
            <password></password>
         </servicePass>
         <!--Optional:-->
         <adom>root</adom>
   </r20:getSystemStatus>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<adom>	If the ADOM field is blank, it is assigned as root.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getSystemStatusResponse>
         <errorMsg>
         <errorCode>0</errorCode>
            <errorMsg>getSystemStatus successfully</errorMsg>
            </errorMsg>
         <platformType>FMG-VM64</platformType>
         <version>v5.0-build0114 130118 (Interim)</version>
         <serialNumber>FMG-VM0A11000137</serialNumber>
         <biosVersion>04000002</biosVersion>
         <hostName>FMG-VM64</hostName>
         <maxNumAdminDomains>1000000000</maxNumAdminDomains>
         <maxNumDeviceGroup>1000000000</maxNumDeviceGroup>
         <adminDomainConf>Enabled</adminDomainConf>
         <fipsMode>Disabled</fipsMode>
   </ns3:getSystemStatusResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • -101: Invalid username, password, or ADOM.
<errorMsg>	
<platformType>	Device model information.
<version>	The firmware version, v5.0-build0114 130118 (interim) for example.
<serialNumber>	The serial number of the device, FMG-VM0A11000137, for example.
<biosVersion>	The BIOS version of the device.
<hostName>	The device host name.
<maxNumAdminDomains>	The maximum number of ADOMs.
<maxNumDeviceGroup>	The maximum number of device groups.
<adminDomainConf>	ADOM mode status.
<fipsMode>	FIPS mode status.

getTaskList

Use this request to get a list of tasks as defined on your unit. Only an administrator with the `Super_User` profile can run this command.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getTaskList>
         <servicePass>
            <password></password>
            <userID>admin</userID>
         </servicePass>
         <adom></adom>
         <taskId>1</taskId>
   </r20:getTaskList>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.

Request Field	Description
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<taskId>	Indicates the task ID number. If the `<waitTask>` was false, then the task ID is displayed.

The response is a series of `<return>` tags, each containing information about a task.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getTaskListResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Get task ID detail successfully</errorMsg>
         </errorMsg>
         <taskList>
            <taskId>1</taskId
            <source>5</source>
            <description>system checkpoint task</description>
            <userID>admin</userID>
            <status>4</status>
            <startTime>2012-09-29T15:18:22Z</startTime
         <deviceList>
            <devName>create system checkpoint</devName>
            <ip>0.0.0.0</ip>
            <status>4</status>
         <message>Create system checkpoint succeed</message>
         <history>
            <name>create system checkpoint</name>
            <percentage>0</percentage>
            <description>task start ...</description>
         </history>
         <history>
            <name>create system checkpoint</name>
            <percentage>5</percentage>
            <description>Lock system succeed</description>
         </history>
         ...
         ...
            </deviceList>
         </taskList>
   </ns3:getTaskListResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Retrieved task ID detail successfully. • -101: Invalid task ID. The task ID is empty or invalid. • -102: The task ID does not exist. • -106: Not enough memory.
<errorMsg>	
<taskList>	XML structure consists of the task ID, source, description, user ID, status, and start time variables.
<taskId>	Indicates the task ID number. If the `<waitTask>` was false, then the task ID is displayed.
<source>	Indicates the source of the task: • 0: Device manager • 1: Security console • 2: Copy global object • 3: Install configuration • 4: Script execution • 5: System checkpoint • 6: Import device policy • 7: Install EMS global policy
<description>	Describes the list.
<userID>	The administrator user name.
<status>	Indicates the status of the task: • 1: running • 2: cancelling • 3: cancelled • 4: done • 5: error • 6: aborting • 7: aborted
<startTime>	Indicates the time the task list started.
<deviceList>	XML structure consists of the device name, IP, status, and description.
<devName>	Name of the device host.

Response Field	Description
<ip>	The device IP address.
<status>	Status of the device.
<message>	Description of the task.
<history>	
<name>	The history name.
<percentage>	Percentage of progress bar of each task that has been applied to the device.
<description>	Description of the history.

getTCLRootFile

Use this request to get information about the TCLRoot file as defined on your unit. Only an administrator with the `Super_User` profile can run this command.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getFileRequest>
         <servicePass>
            <password></password>
            <userID>admin</userID>
         </servicePass>
         <fileName></fileName>
         <fileOffset>1</fileOffset>
         <fileMaxLen>10</fileMaxLen>
         <fileEncode>0</fileEncode>
   </r20:getFileRequest>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.

Request Field	Description
<filename>	Shows the name of the file. Note: the file name cannot start with a | or a ~ character.
<fileOffset>	Indicates the starting point in the receiving file. This must be a positive number and must be smaller that the file.
<fileMaxLen>	Indicates the maximum size of the received file. Must be a positive number.
<fileEncode>	Indicates the encoding method of the receiving file: • 0: base64 • 1: hexadecimal base • 2: raw data

The response is a series of <return> tags, each containing information about the file.

Example response:

If the task is successful, you will get a message stating that. It the task is not successful, for example a file called root does not exist, you will get a message similar to the one below:

```
    <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
          <faultcode>SOAP-ENV:Client</faultcode>
          <faultstring>File does not exist</faultstring>
          <detail>root</detail>
    </SOAP-ENV:Fault>
</SOAP-ENV:Body>
```

Response Field	Description
<fileName>	Shows the name of the file. Note: the file name cannot start with a | or a ~ character.
<fileOffset>	Indicates the starting point in the receiving file. This must be a positive number and must be smaller that the file.
<fileMaxLen>	Indicates the maximum size of the received file. Must be a positive number.
<fileEncode>	Indicates the encoding method of the receiving file: • 0: base64 • 1: hexadecimal base • 2: raw data

importPolicy

Use this request to import a policy.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:importPolicy>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <adomName>?</adomName>
         <!--Optional:-->
         <adomOid>?</adomOid>
         <!--Optional:-->
         <devName>?</devName>
         <!--Optional:-->
         <devId>?</devId>
         <!--Optional:-->
         <vdomName>?</vdomName>
         <!--Optional:-->
         <vdomId>?</vdomId>
         <!--Optional:-->
         <policyPackageName>?</policyPackageName>
   </r20:importPolicy>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<adomName>	The ADOM name.
<adomOid>	The ADOM identifier.
<devName>	The device name.
<devId>	The device ID. This is the primary device identifier.
<vdomName>	The name of the VDOM.
<vdomId>	The VDOM identifier.
<policyPackageName>	The policy package name.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:importPolicyResponse>
         <adomName>root</adomName>
         <adomOid>3</adomOid>
         <devName>FGT60C3G06500185</devName>
         <devId>129</devId>
         <vdomName>root</vdomName>
         <vdomId>3</vdomId>
         <report>Start to import config from device(129) vdom(root) to adom(3)
   "firewall service category",SUCCESS,"(name=General, oid=331, DUPLICATE)"
   "firewall schedule recurring",SUCCESS,"(name=always, oid=685, DUPLICATE)"
   "firewall address",SUCCESS,"(name=all, oid=322, DUPLICATE)"
   "firewall service custom",SUCCESS,"(name=ALL, oid=595, DUPLICATE)"
   "webfilter urlfilter",SUCCESS,"(name="gsdg", oid=806, DUPLICATE)"
   "webfilter ftgd-local-cat",SUCCESS,"(name=custom1, oid=341, DUPLICATE)"
   "webfilter ftgd-local-cat",SUCCESS,"(name=custom2, oid=342, DUPLICATE)"
   "webfilter ftgd-local-cat",SUCCESS,"(name=ls, oid=801, DUPLICATE)"
   "webfilter ftgd-local-rating",SUCCESS,"(name=hjh, oid=811, DUPLICATE)"
   "application list",SUCCESS,"(name=client-reputation, oid=369, DUPLICATE)"
   "firewall profile-protocol-options",SUCCESS,"(name=default, oid=686, DUPLICATE)"
   "webfilter profile",SUCCESS,"(name=client-reputation, oid=518, DUPLICATE)"
   "firewall policy",SUCCESS,"(name=ID:1 (#1), oid=804)"</report>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Ended importing policies from adom:3 dev:129</errorMsg>
         </errorMsg>
   </ns3:importPolicyResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<adomName>	The ADOM name.
<adomOid>	The ADOM object identifier.
<devName>	The device name
<devId>	The device ID. This is the primary device identifier.
<vdomName>	The name of the VDOM.
<vdomId>	The VDOM identifier.
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.

Response Field	Description
<errorCode>	Error code and message details: • 0: Ended importing policies from ADOM device. • -101: Cannot find the ADOM by provided name or OID. Cannot find the device by provided name or OID. Cannot find the VDOM by provided name or ID. The provided policy package already exists. One provided policy package name for more than one VDOM. • -104: The user does not have access to device VDOM in ADOM. Searching policies error for ADOM device VDOM. Fetch summary file error for ADOM device. Creating context error for device. Update zone or add zone mappings error for ADOM device VDOM. No policies found for ADOM device, VDOM. Selecting package name error for ADOM device VDOM. Searching policy objects error for ADOM device VDOM. Fetch summary file error for ADOM device. Importing policies error for ADOM device VDOM. Fetch report file error for ADOM device.
<errorMsg>	

listRevisionId

Use this request to get a list of the revisions as defined on your unit. Only an administrator with the `Super_User` profile can run this command.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:listRevisionId>
         <servicePass>
            <password>?</password>
            <userID>admin</userID>
         </servicePass>
         <devId>?</devId>
         <serialNumber>?</serialNumber>
         <revName>?</revName>
   </r20:listRevisionId>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.

Request Field	Description
<devid>	The device ID. This is the primary device identifier.
<serialNumber>	Serial number of device, FGT60C3G06500185, for example.
<revName>	The name of the revision file.

The response is a series of <return> tags, each containing information about the revisions.

Example response:

If the task is successful, you will get the revision information. If the task is unsuccessful, you will get an error message.

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Total match revision ID(s) with revision name. Total match revision ID(s). • -102: The device is in backup mode. Cannot get revision ID(s) from revision name at device ID. Cannot get all revision ID(s) at device ID.
<errorMsg>	
<devId>	The device ID. This is the primary device identifier.

retrieveConfig

Use this request to retrieve the latest running configuration from the FortiGate unit, and to create a new revision as defined on your unit. Only an administrator with the `Super_User` profile can run this command.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:retrieveConfig>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <devId>?</devId>
         <!--Optional:-->
         <serialNumber>?</serialNumber>
```

```
            <!--Optional:-->
            <newRevName>?</newRevName>
      </r20:retrieveConfig>
   </soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<devid>	The device ID. This is the primary device identifier.
<serialNumber>	Serial number of device, FGT60C3G06500185, for example.
<newRevName>	The new name of the revision file. The length should be from 1 to 49 characters.

If the task is successful you will get the task ID.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:retrieveConfigResponse>
   <errorMsg>
         <errorCode>-104</errorCode>
         <errorMsg>run retrieveConfig task failed</errorMsg>
   </errorMsg>
   </ns3:retrieveConfigResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Read task ID to get retrieve configuration result. • -102: The device is in backup mode. Cannot get device information from the group ID. • -101: The total devices/groups number exceeds limit. The new revision name length should be 1 - 49 characters. The new revision name count does not match with device count. • -104: The run retrieve configuration task failed. Retrieve configuration from device ID/groupID failed.
<errorMsg>	

revertConfig

Use this request to revert to the previous configuration on your unit. Only an administrator with the `Super_User` profile can run this command.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:revertConfig>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <devId>?</devId>
         <!--Optional:-->
         <serialNumber>?</serialNumber>
         <revId>?</revId>
   </r20:revertConfig>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: • Enter the administrator password. • Leave field blank for no password.
<devid>	The device ID. This is the primary device identifier.
<serialNumber>	Serial number of device, FGT60C3G06500185, for example.
<revId>	The revision ID number.

The response indicates if the configuration reverted successfully or not.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <r20:revertConfig>
   <errorMsg>
         <errorCode>-104</errorCode>
         errorMsg>Revert revision 1 on deviceId 661 successful></errorMs
   </r20:revertConfig>
   </errorMsg>
```

```
    </r20:revertConfig>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Revert to revision on device ID successfully. • -101: Invalid revision ID value. • -102: The device is in backup mode. Revision does not exist on device ID. • -104 : Revert revision on device ID failed.
<errorMsg>	

FortiAnalyzer XML API elements

getFazConfig

Use this request to get the FortiAnalyzer configuration.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getFazConfig>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password></password>
      </servicePass>
   </r20:getFazConfig>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getFazConfigResponse>
   <errorMsg>
   <errorCode>0</errorCode>
   <errorMsg>getFazConfig successfully</errorMsg>
   </errorMsg>
   <config>#config-version=FAZVM-5.0-FW-build115-130121
   config system global
      set adom-mode normal
```

```
   set hostname "FMG-VM"
end
config system interface
   edit "port1"
      set ip 172.16.106.254 255.255.255.0
      set allowaccess ping https ssh http webservice
      set serviceaccess fgtupdates webfilter-antispam webfilter antispam
         config ipv6
         end
   next
   edit "port2"
      set ip 1.2.2.2 255.255.255.0
      set allowaccess ping https ssh http webservice
      set serviceaccess fgtupdates webfilter-antispam webfilter antispam
         config ipv6
         end
   next
   edit "port3"
         config ipv6
         end
   next
   edit "port4"
         config ipv6
         end
   next
end
config system snmp sysinfo
end
config system route
   edit 1
      set device "port1"
      set gateway 172.16.106.1
   next
end
config system dns
   set primary 208.91.112.53
   set secondary 208.91.112.63
end
config system ha
end
config system ntp
config ntpserver
   edit 1
      set server "ntp1.fortinet.net"
   next
   end
      set status enable
      set sync_interval 1
end
config system backup all-settings
end
config system metadata admins
   edit "Contact Email"
      set importance optional
   next
   edit "Contact Phone"
      set importance optional
```

```
    next
  end
  config system admin profile
    edit "Restricted_User"
      set description "Restricted user profiles have no System Privileges enabled, and
          have read-only access for all Device Privileges."
      set device-manager read
      set device-config read
      set device-profile read
      set policy-objects read
      set deploy-management read
      set config-retrieve read
      set term-access read
      set adom-policy-packages read
      set adom-policy-objects read
      set vpn-manager read
      set realtime-monitor read
      set forticonsole read
      set consistency-check read
      set faz-management read
      set log-viewer read
      set report-viewer read
    next
    edit "Standard_User"
      set description "Standard user profiles have no System Privileges enabled, but
          have read/write access for all Device Privileges."
      set adom-switch read-write
      set global-policy-packages read-write
      set global-objects read-write
      set device-manager read-write
      set device-config read-write
      set device-op read-write
      set device-profile read-write
      set policy-objects read-write
      set deploy-management read-write
      set config-retrieve read-write
      set term-access read-write
      set adom-policy-packages read-write
      set adom-policy-objects read-write
      set vpn-manager read-write
      set realtime-monitor read-write
      set forticonsole read-write
      set consistency-check read-write
      set faz-management read-write
      set log-viewer read-write
      set report-viewer read-write
    next
    edit "Super_User"
      set description "Super user profiles have all system and device privileges
          enabled."
      set system-setting read-write
      set adom-switch read-write
      set global-policy-packages read-write
      set global-objects read-write
      set assignment read-write
      set read-passwd read-write
      set device-manager read-write
```

```
            set device-config read-write
            set device-op read-write
            set device-profile read-write
            set policy-objects read-write
            set deploy-management read-write
            set config-retrieve read-write
            set term-access read-write
            set adom-policy-packages read-write
            set adom-policy-objects read-write
            set vpn-manager read-write
            set realtime-monitor read-write
            set forticonsole read-write
            set consistency-check read-write
            set faz-management read-write
            set log-viewer read-write
            set report-viewer read-write
        next
        edit "Package_User"
            set description "Package user profile have read/write policy package and objects
                privileges enabled, and have read-only access for system and others
                privileges."
            set system-setting read
            set adom-switch read
            set global-policy-packages read-write
            set global-objects read-write
            set assignment read
            set read-passwd read
            set device-manager read-write
            set device-config read-write
            set device-op read-write
            set device-profile read-write
            set policy-objects read-write
            set deploy-management read-write
            set config-retrieve read
            set term-access read
            set adom-policy-packages read-write
            set adom-policy-objects read-write
            set vpn-manager read-write
            set realtime-monitor read
            set forticonsole read
            set consistency-check read
            set faz-management read
            set log-viewer read
            set report-viewer read
        next
    end
    config system certificate ca
    end
    config system certificate local
    end
    config system password-policy
    end
    config system admin user
        edit "admin"
            set trusthost2 0.0.0.0 0.0.0.0
            set trusthost3 127.0.0.1 255.255.255.255
            set ipv6_trusthost2 ::/0
```

```
      set ipv6_trusthost3 ::1/128
      set profileid "Super_User"
      set adom "all_adoms"
      set policy-package "all_policy_packages"
config dashboard
   edit 1
      set name "System Information"
      set column 1
      set refresh-interval 0
      set tabid 1
      set widget-type sysinfo
      next
   edit 2
      set name "System Resources"
      set column 1
      set refresh-interval 0
      set tabid 1
      set widget-type sysres
      set res-view-type real-time
      next
   edit 3
      set name "License Information"
      set column 2
      set refresh-interval 0
      set tabid 1
      set widget-type licinfo
      next
   edit 4
      set name "Unit Operation"
      set column 2
      set refresh-interval 0
      set tabid 1
      set widget-type sysop
      next
   edit 5
      set name "Alert Message Console"
      set column 2
      set refresh-interval 0
      set tabid 1
      set widget-type alert
      set num-entries 0
      next
   end
   next
end
config system admin setting
end
config system alertemail
end
config system mail
   edit "mail.fortinet.com"
      set auth enable
      set passwd ENC
          26ITYiEXHPFvx8y3vZqI4PPt2dH0OXAWPB3sVNcK+2nPTGyeRN1FMB+hJIlyHsyzchBxBmA2EMZEj
          y4gR5vBnYiufPp2Q5rcGhSAYqGQ2zMSt79R
      set user "jsmith@fortinet.com"
      next
```

```
   end
config system alert-console
end
config system log fortianalyzer
end
config system locallog disk setting
end
config system locallog disk filter
end
config system locallog memory setting
end
config system locallog memory filter
end
config system locallog fortianalyzer setting
end
config system locallog fortianalyzer filter
end
config system locallog syslogd setting
end
config system locallog syslogd filter
end
config system locallog syslogd2 setting
end
config system locallog syslogd2 filter
end
config system locallog syslogd3 setting
end
config system locallog syslogd3 filter
end
config system fips
end
config fmupdate av-ips fgt server-override
end
config fmupdate av-ips fct server-override
end
config fmupdate web-spam fgt server-override
end
config fmupdate web-spam fct server-override
end
config fmupdate av-ips push-override
end
config fmupdate av-ips push-override-to-client
end
config fmupdate web-spam poll-frequency
end
config fmupdate av-ips web-proxy
end
config fmupdate web-spam web-proxy
end
config fmupdate fct-services
end
config fmupdate av-ips advanced-log
end
config fmupdate av-ips update-schedule
end
config fmupdate analyzer virusreport
end
```

```
        config fmupdate service
        end
        config fmupdate publicnetwork
        end
        config fmupdate disk-quota
        end
        config fmupdate server-access-priorities
        end
        config fmupdate web-spam fgd-setting
        end
        config fmupdate web-spam fgd-log
        end
        config fmupdate custom-url-list
        end
        config fmupdate device-version
        end
        config fmupdate deployment
        end
        config fmupdate server-override-status
        end
        config fmupdate multilayer
        end
        config fmupdate support-pre-fgt43
        end
        config system dm
        end
        config system log settings
        config rolling-regular
        end
        end
           config system sql
              set start-time 09:37 2013/01/18
           end
        </config>
        </ns3:getFazConfigResponse>
    </SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • -101: Invalid username or password. • -102: Cannot allocate temp file. Cannot create configuration file. Cannot open file. • -106: Not enough memory.
<errorMsg>	
<config>	The device configuration.

setFazConfig

Use this request to set the FortiAnalyzer configuration. You can set either partial or full configuration.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:setFazConfig>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password></password>
      </servicePass>
      <!--Optional:-->
      <adom>root</adom>
      <!--Optional:-->
      <config>
   config system global
      set adom-mode normal
      set hostname "FMG-VM"
   end
   config system interface
      edit "port1"
         set ip 172.16.106.254 255.255.255.0
         set allowaccess ping https ssh http webservice
         set serviceaccess fgtupdates webfilter-antispam webfilter antispam
            config ipv6
            end
      next
      edit "port2"
         set ip 1.2.2.2 255.255.255.0
         set allowaccess ping https ssh http webservice
         set serviceaccess fgtupdates webfilter-antispam webfilter antispam
            config ipv6
            end
      next
      edit "port3"
            config ipv6
            end
      next
      edit "port4"
            config ipv6
            end
      next
   end
   config system snmp sysinfo
   end
   config system route
      edit 1
         set device "port1"
         set gateway 172.16.106.1
      next
   end
   config system dns
      set primary 208.91.112.53
      set secondary 208.91.112.63
   end
   config system ha
```

```
end
config system ntp
config ntpserver
   edit 1
      set server "ntp1.fortinet.net"
   next
   end
      set status enable
      set sync_interval 1
end
config system backup all-settings
end
config system metadata admins
   edit "Contact Email"
      set importance optional
   next
   edit "Contact Phone"
      set importance optional
   next
end
config system admin profile
   edit "Restricted_User"
      set description "Restricted user profiles have no System Privileges enabled, and
          have read-only access for all Device Privileges."
      set device-manager read
      set device-config read
      set device-profile read
      set policy-objects read
      set deploy-management read
      set config-retrieve read
      set term-access read
      set adom-policy-packages read
      set adom-policy-objects read
      set vpn-manager read
      set realtime-monitor read
      set forticonsole read
      set consistency-check read
      set faz-management read
      set log-viewer read
      set report-viewer read
   next
   edit "Standard_User"
      set description "Standard user profiles have no System Privileges enabled, but
          have read/write access for all Device Privileges."
      set adom-switch read-write
      set global-policy-packages read-write
      set global-objects read-write
      set device-manager read-write
      set device-config read-write
      set device-op read-write
      set device-profile read-write
      set policy-objects read-write
      set deploy-management read-write
      set config-retrieve read-write
      set term-access read-write
      set adom-policy-packages read-write
      set adom-policy-objects read-write
```

```
            set vpn-manager read-write
            set realtime-monitor read-write
            set forticonsole read-write
            set consistency-check read-write
            set faz-management read-write
            set log-viewer read-write
            set report-viewer read-write
        next
        edit "Super_User"
            set description "Super user profiles have all system and device privileges
                enabled."
            set system-setting read-write
            set adom-switch read-write
            set global-policy-packages read-write
            set global-objects read-write
            set assignment read-write
            set read-passwd read-write
            set device-manager read-write
            set device-config read-write
            set device-op read-write
            set device-profile read-write
            set policy-objects read-write
            set deploy-management read-write
            set config-retrieve read-write
            set term-access read-write
            set adom-policy-packages read-write
            set adom-policy-objects read-write
            set vpn-manager read-write
            set realtime-monitor read-write
            set forticonsole read-write
            set consistency-check read-write
            set faz-management read-write
            set log-viewer read-write
            set report-viewer read-write
        next
        edit "Package_User"
            set description "Package user profile have read/write policy package and objects
                privileges enabled, and have read-only access for system and others
                privileges."
            set system-setting read
            set adom-switch read
            set global-policy-packages read-write
            set global-objects read-write
            set assignment read
            set read-passwd read
            set device-manager read-write
            set device-config read-write
            set device-op read-write
            set device-profile read-write
            set policy-objects read-write
            set deploy-management read-write
            set config-retrieve read
            set term-access read
            set adom-policy-packages read-write
            set adom-policy-objects read-write
            set vpn-manager read-write
            set realtime-monitor read
```

```
      set forticonsole read
      set consistency-check read
      set faz-management read
      set log-viewer read
      set report-viewer read
   next
end
config system certificate ca
end
config system certificate local
end
config system password-policy
end
config system admin user
   edit "admin"
      set trusthost2 0.0.0.0 0.0.0.0
      set trusthost3 127.0.0.1 255.255.255.255
      set ipv6_trusthost2 ::/0
      set ipv6_trusthost3 ::1/128
      set profileid "Super_User"
      set adom "all_adoms"
      set policy-package "all_policy_packages"
config dashboard
   edit 1
      set name "System Information"
      set column 1
      set refresh-interval 0
      set tabid 1
      set widget-type sysinfo
      next
   edit 2
      set name "System Resources"
      set column 1
      set refresh-interval 0
      set tabid 1
      set widget-type sysres
      set res-view-type real-time
      next
   edit 3
      set name "License Information"
      set column 2
      set refresh-interval 0
      set tabid 1
      set widget-type licinfo
      next
   edit 4
      set name "Unit Operation"
      set column 2
      set refresh-interval 0
      set tabid 1
      set widget-type sysop
      next
   edit 5
      set name "Alert Message Console"
      set column 2
      set refresh-interval 0
      set tabid 1
```

```
      set widget-type alert
      set num-entries 0
      next
   end
   next
end
config system admin setting
end
config system alertemail
end
config system mail
   edit "mail.fortinet.com"
      set auth enable
      set passwd ENC
          26ITYiEXHPFvx8y3vZqI4PPt2dH0OXAWPB3sVNcK+2nPTGyeRN1FMB+hJIlyHsyzchBxBmA2EMZEj
          y4gR5vBnYiufPp2Q5rcGhSAYqGQ2zMSt79R
      set user "jsmith@fortinet.com"
      next
   end
config system alert-console
end
config system log fortianalyzer
end
config system locallog disk setting
end
config system locallog disk filter
end
config system locallog memory setting
end
config system locallog memory filter
end
config system locallog fortianalyzer setting
end
config system locallog fortianalyzer filter
end
config system locallog syslogd setting
end
config system locallog syslogd filter
end
config system locallog syslogd2 setting
end
config system locallog syslogd2 filter
end
config system locallog syslogd3 setting
end
config system locallog syslogd3 filter
end
config system fips
end
config fmupdate av-ips fgt server-override
end
config fmupdate av-ips fct server-override
end
config fmupdate web-spam fgt server-override
end
config fmupdate web-spam fct server-override
end
```

```
      config fmupdate av-ips push-override
      end
      config fmupdate av-ips push-override-to-client
      end
      config fmupdate web-spam poll-frequency
      end
      config fmupdate av-ips web-proxy
      end
      config fmupdate web-spam web-proxy
      end
      config fmupdate fct-services
      end
      config fmupdate av-ips advanced-log
      end
      config fmupdate av-ips update-schedule
      end
      config fmupdate analyzer virusreport
      end
      config fmupdate service
      end
      config fmupdate publicnetwork
      end
      config fmupdate disk-quota
      end
      config fmupdate server-access-priorities
      end
      config fmupdate web-spam fgd-setting
      end
      config fmupdate web-spam fgd-log
      end
      config fmupdate custom-url-list
      end
      config fmupdate device-version
      end
      config fmupdate deployment
      end
      config fmupdate server-override-status
      end
      config fmupdate multilayer
      end
      config fmupdate support-pre-fgt43
      end
      config system dm
      end
      config system log settings
      config rolling-regular
      end
      end
         config system sql
            set start-time 09:37 2013/01/18
         end
         </config>
      </r20:setFazConfig>
   </soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	The ADOMs for which you want to set the configuration.
<config>	The configuration content to be sent.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:setFazConfigResponse>
      <errorMsg>
         <errorCode>0</errorCode>
         <errorMsg>setFazConfig failed</errorMsg>
      </errorMsg>
      <cliError>command parse error before 'webfilter'
      (port1)#</cliError>
      <errorLineNumber>10</errorLineNumber>
   </ns3:setFazConfigResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • -101: Invalid username, password, or ADOM. Cannot get configuration value. • -104: Cannot set configuration.
<errorMsg>	
<errorLineNumber>	The line number where the error occurs.

runFazReport

Use this request to run a report through web services. You need to input the schedule name of the report. In v5.2 Patch Release 2 or later you can add filters to support per user reports. `runFazReport` supports up to 10k filters.

Example request:

```
<soapenv:Header/>
```

```
<soapenv:Body>
   <r20:runFazReport>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password></password>
      </servicePass>
      <!--Optional:-->
      <adom>root</adom>
      <!--Optional:-->
      <reportTemplate>12</reportTemplate>
      <filter>user=USER00001</filter>
      <filter>user=USER00002</filter>
   </r20:runFazReport>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	The ADOMs for which you want to run a report against.
<reportTemplate>	The name of the report template.
<filter>	Add filters to create per-user reports.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:runFazReportResponse>
      <errorMsg>
         <errorCode>0</errorCode>
         <errorMsg>runFazReport successfully</errorMsg>
      </errorMsg>
      <reportTemplate>12</reportTemplate>
   </ns3:runFazReportResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.

Response Field	Description
<errorCode>	Error code and message details: • -101: Invalid username, password, or ADOM. Cannot get report schedule name. • -104: Cannot get report schedule name from SQL.
<errorMsg>	
<reportTemplate>	The name of the report template.

getFazGeneratedReports

Use this request to get a completed historical report. To use this command, you need to input the report name, report date, compression method in the request.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getFazGeneratedReport>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password></password>
      </servicePass>
      <!--Optional:-->
      <adom>root</adom>
      <!--Optional:-->
      <reportDate>2013_01_24</reportDate>
      <!--Optional:-->
      <reportName>S-4_t4-2013-01-24-1611</reportName>
      <!--Optional:-->
      <compression>tar</compression>
   </r20:getFazGeneratedReport>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	The ADOMs for which you want to get a report from.
<reportDate>	The report generation date; in the format YYYY_MM_DD.

Request Field	Description
<reportName>	The generated report name. For example, S-schedule-utm-reports_t1-2013-01-24-1022.
<compression>	The compression type of the report that will be returned by this command. Compression options include: • 0: tar • 1: gzip

The report data returned in the response message is base64 encoded binary data. You need to decode it and then decompress it to get the report files.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getFazGeneratedReportResponse>
      <errorMsg>
         <errorCode>0</errorCode>
         <errorMsg>getFazGeneratedReport successfully</errorMsg>
      </errorMsg>
      <reportName>S-4_t4-2013-01-24-1611</reportName>
      <size>71680</size>
      <fazReportData>
      <reportContent>
      </fazReportData>
   </ns3:getFazGeneratedReportResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • -101: Invalid username, password, or ADOM. Cannot get report name. Cannot get report date. • -104: Cannot find report name. Cannot find file in directory. Cannot read file in directory. • -106: Not enough memory.
<errorMsg>	
<reportName>	The generated report name. For example, S-schedule-utm-reports_t1-2013-01-24-1022.
<size>	The generated report size.
<fazReportData>	Report content data will be displayed under this element.
<reportContent>	Contains the actual report data. The data is base64 encoded, you need to decode the data before use.

searchFazLog

Use this request to provide raw logs in FortiAnalyzer per conditions set in the request. You need to input the log format, device name, log type, search criteria, start index, and maximum return value in the request message body.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:searchFazLog>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password></password>
      </servicePass>
      <!--Optional:-->
      <adom>root</adom>
      <!--Optional:-->
      <content>logs</content>
      <!--Optional:-->
      <format>rawFormat</format>
      <!--Optional:-->
      <deviceName>FG200B-1</deviceName>
      <logType>traffic</logType>
      <!--Optional:-->
      <searchCriteria>srcip=10.0.0.1</searchCriteria>
      <maxNumMatches>30</maxNumMatches>
      <startIndex>1</startIndex>
      <checkArchive>0</checkArchive>
      <!--Optional:-->
      <DLPArchiveType>1</DLPArchiveType>
      <!--Optional:-->
      <compression>tar</compression>
   </r20:searchFazLog>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	The ADOMs for which you want to search logs.

Request Field	Description
<content>	The log contents you want to search. Log content options: • 0: Logs • 1: Content logs • 2: Local logs
<format>	The log formats to display. Log format options: • 0: Raw • 1: CSV
<deviceName>	The device name you want to search logs from.
<logType>	Type of logs you want to search. Log type options: • 0: Event • 1: Traffic • 2: Attack • 3: Antivirus • 4: Web logs • 5: IM • 6: Email • 7: Content • 8: History • 9: Generic • 10: VoIP • 11: DLP • 12: Application Control • 13: Network Scanning
<searchCriteria>	The search criteria used to search logs. For example, vd-root.
<maxNumMatches>	The maximum number of matches to display from the search results.
<startIndex>	The start index of the matched logs.
<checkArchive>	This variable is not used. Always set the value to 0.
<DLPArchiveType>	The DLP archive type. DLP archive type options: • 0: Web • 1: Email • 2: FTP • 3: IM • 4: MMS • 5: Quarantine • 6: IPS

Request Field	Description
<compression>	The compression type of the report that will be returned by this command. Compression options: • 0: tar • 1: gzip

The response will contain the logs that match the criteria specified in the request.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:searchFazLogResponse>
      <errorMsg>
         <errorCode>0</errorCode>
         <errorMsg>searchFazLog successfully</errorMsg>
      </errorMsg>
      <totalResultsFound>300</totalResultsFound>
      <matchesReturned>16</matchesReturned>
      <startIndex>1</startIndex>
      <logs>
         <data>
            <logEntry>date=2013-01-25 time=09:50:58 itime=1359107458 logid=2222222222
                type=ips subtype=status=accept level=level40 devid=FG200B0000000001
                policyid=10000 sessionid=10000 attackid=10000 severity=severity
                profile=profile40 sensor=sensor40 srcip=10.0.0.1 dstip=10.0.0.1
                srcport=1000 icmpid=icmpid40 dstport=1000 icmptype=icmpty icmpcode=icmpco
                srcintf=srcintf40 dstintf=dstintf40 proto=0 service=smtp user=user1
                group=group40 ref=ref40 count=10000 incidentserialno=10000 msg=msg40 vd=vd1
                identidx=10000 profiletype=profiletype40 profilegroup=prof
                attackname=attackname40 direction=10000 dstname=dstname40 srcname=srcname40
                agent=agent40 osname=osname40 osversion=osversion40 unauthuser=unauthuser40
                unauthusersource=unauthusersource40 eventtype=eventtype40</logEntry>
         </data>
      </logs>
   </ns3:searchFazLogResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • -101: Invalid username, password, or ADOM. Cannot get device name. Cannot get search criteria. Incorrect DLP archive type. • -102: Cannot find device name on system. • -104: Cannot find logs with criteria. • -106: Not enough memory.
<errorMsg>	
<totalResultsFound>	The total number of logs found.

Response Field	Description
<matchesReturned>	The total number of logs which matched the search criteria.
<startIndex>	The start index in the request.
<logs>	Log data will be displayed under this element.
<logEntry>	Displays log data.

getFazArchive

Use this request to get a FortiAnalyzer archive file. You need to input the device ID, archive type and the archive file name.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getFazArchive>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password></password>
      </servicePass>
      <!--Optional:-->
      <adom>root</adom>
      <!--Optional:-->
      <devId>FG200B0000000001</devId>
      <type>IPS</type>
      <!--Optional:-->
      <fileName>50005:0</fileName>
      <zipPassword></zipPassword>>
   </r20:getFazArchive>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	The ADOMs for which you want to get archives from.
<devId>	The device ID you want to get archives from.

Request Field	Description
<type>	The archive type. Archive type options: • 0: Web • 1: Email • 2: FTP • 3: IM • 4: MMS • 5: Quarantine • 6: IPS
<fileName>	The archive file name. You can check the name under *Log View* > *Archive*.
<zipPassword>	The password set for the zip file.
<filelist>	The archive file list.
<filename>	The archive file name will be displayed under this element.
<data>	The archive file content data. The data is base64 encoded, you need to decode the data before use.

The response will contain the binary data if the archive file in a base64 encoded message.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getFazArchiveResponse>
      <errorMsg>
         <errorCode>0</errorCode>
         <errorMsg>getFazArchive successfully</errorMsg>
      </errorMsg>
      <fileList>
      <fileName>50005:0</fileName>
         <data>
         =</data>
         <error>None</error>
      </fileList>
   </ns3:getFazArchiveResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.

Response Field	Description
<errorCode>	Error code and message details: • -101: Invalid username, password, or ADOM. Cannot get device ID. Cannot get file name. Cannot get type. Cannot get checksum. Cannot get content filename. • -104: Cannot get content archive. Get FortiAnalyzer archive failed, no such file name. Get FortiAnalyzer archive failed, error reading file name. • -106: Not enough memory.
<errorMsg>	
<filelist>	The archive file list.
<filename>	The archive file name will be displayed under this element.
<data>	The archive file content data. The data is base64 encoded, you need to decode the data before use.

listFazGeneratedReports

Use this request to list FortiAnalyzer generated reports.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:listFazGeneratedReports>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password></password>
      </servicePass>
      <!--Optional:-->
      <adom>root</adom>
      <!--Optional:-->
      <startDate>2013-02-04T00:00:00</startDate>
      <!--Optional:-->
      <endDate>2013-02-05T00:00:00</endDate>
   </r20:listFazGeneratedReports>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.

Request Field	Description
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	The ADOMs for which you want to list a generated reports.
<startDate>	The report start date.
<endDate>	The report end date.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:listFazGeneratedReportsResponse>
      <errorMsg>
         <errorCode>0</errorCode>
         <errorMsg>listFazGeneratedReports successfully</errorMsg>
      </errorMsg>
      <totalNumberExists>48</totalNumberExists>
      <reportList>
         <reportName>S-schedule-utm-reports_t1-2013-02-04-0000</reportName>
         <startTime>2013-02-04T08:00:03Z</startTime>
         <endTime>2013-02-04T08:02:44Z</endTime>
         <reportProgressPercent>100</reportProgressPercent>
         <size>52122</size>
         <formats>PH</formats>
      </reportList>
   </ns3:listFazGeneratedReportsResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • -101: Invalid username, password, or ADOM. No reports are available. • -104: Cannot get report counts. • -106: Not enough memory.
<errorMsg>	
<totalNumberExists>	All available reports in FortiAnalyzer.
<reportList>	XML structure consists of report name, start time, end time, report progress, size, and format variables.

Response Field	Description
<reportName>	The generated report name. For example, S-schedule-utm-reports_t1-2013-01-24-1022.
<startTime>	Indicates the time the report started.
<endTime>	Indicates the time the report ended.
<reportProgressPercent>	Report running progress; 0 to 100%.
<size>	The generated report size.
<formats>	The report format: • P: PDF • H: HTML • T: TXT

removeFazArchive

Use this command to remove a FortiAnalyzer archive.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:removeFazArchive>
      <!--Optional:-->
      <servicePass>
         <!--Optional:-->
         <userID>admin</userID>
         <!--Optional:-->
         <password></password>
      </servicePass>
      <!--Optional:-->
      <adom>root</adom>
      <!--Optional:-->
      <devId>FG200B0000000001</devId>
      <!--Optional:-->
      <type>IPS</type>
      <!--Optional:-->
      <fileName>50008:0</fileName>
      <!--Optional:-->
      <checksum>?</checksum>
   </r20:removeFazArchive>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.

Request Field	Description
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<adom>	The ADOM for which you want to remove the FortiAnalyzer archive.
<devId>	The device ID. This is the primary device identifier.
<type>	The archive type: • 0: Web • 1: Email • 2: FTP • 3: IM • 4: MMS • 5: Quarantine • 6: IPS
<fileName>	Shows the name of the file. Note: the file name cannot start with a | or a ~ character.
<checksum>	Checksum is used when the type is Quarantine. Checksum is used instead of filename.

The response indicates if the request was successful or if it failed.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:removeFazArchiveResponse>
      <errorMsg>
         <errorCode>0</errorCode>
         <errorMsg>removeFazArchive successfully</errorMsg>
      </errorMsg>
   </ns3:removeFazArchiveResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • -101: Invalid username, password, or ADOM. Cannot get the device ID. Cannot get the file name. Cannot get the type. Cannot get the checksum. • -104: Cannot delete content archive file.
<errorMsg>	

Script XML API elements

You can upload scripts to your unit and execute them on your database or on a managed device. If your scripts make configuration changes to the database, you can install the changes onto the affected devices.

createScript

Use this request to upload a script to your unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:createScript>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <adom>?</adom>
         <!--Optional:-->
         <isGlobal>?</isGlobal>
         <!--Optional:-->
         <name>?</name>
         <!--Optional:-->
         <type>?</type>
         <!--Optional:-->
         <description>?</description>
         <!--Optional:-->
         <content>?</content>
         <!--Optional:-->
         <target>?</target>
         <!--Optional:-->
         <overwrite>?</overwrite>
   </r20:createScript>
</soapenv:Body>
```

Request Field	Description
<servicepass>	XML structure consists of username and password variables.

Request Field	Description
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<isGlobal>	Set for global script.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<name>	The script name.
<type>	The script type.
<description>	Brief script description (optional).
<content>	The script content.
<target>	The script target: *Device Database*, *Remote Device*, or *ADOM Database*.
<overwrite>	Overwrite value options: • 1: to overwrite an existing script of that name. • 0: to keep the name of the script.

The response is a `<return>` value of 0 if successful, 1 if not. If `<overwrite>` was 0, `createScript` can fail because there is already a script of that name on your unit.

Example response: (script created)

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:createScriptResponse>
         <return>0</return>
   </ns3:createScriptResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<return>	Return codes: • 0: Created script successfully. • 1: Create script failed.

deleteScript

Use this request to delete a script from your unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:deleteScript>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <name>?</name>
         <!--Optional:-->
         <type>?</type>
   </r20:deleteScript>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<name>	The name of the script to delete.
<type>	The script type, e.g. CLI, TCL, CLIGROUP.

If the script is found and deleted, the response is empty. If the script could not be found, Web Services returns an error message.

Example response: (script was deleted)

```
<ns3:deleteScriptResponse/>
```

Example response: (script not found)

```
<faultcode>SOAP-ENV:Client</faultcode>
<faultstring>script xml_script1 is not found</faultstring>
<detail>
<error xmlns="http://localhost/">script xml_script1 is not found</error>
</detail>
```

Response Field	Description
<faultcode>	These are considered as generic SOAP errors. But there are cases that errors from the application level also return inside `<SOAP-ENV:Fault>` envelop. These errors are free-style, there are no error code associated.
<faultstring>	

getScript

Use this command to retrieve a script from your unit. This is a way to verify the contents of the script. Also, you could modify the script and use the `createScript` request to update the script on your unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getScript>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <name>?</name>
         <!--Optional:-->
         <type>?</type>
   </r20:getScript>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<name>	The script name.
<type>	The script type, e.g. CLI, TCL, CLIGROUP.

The response is a return tag that includes the script content and information about the script.

Example response:

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:getScriptResponse>
   <return>
         <content>
         ... script ...
         </content>
         <description>Generated by XML API</description>
         <name>script1</name>
         <oid>14</oid>
   </return>
   </ns3:getScriptResponse>
```

```
</SOAP-ENV:Body>
```

Response Field	Description
<content>	XML structure consists of description, name, and object identifier variables.
<description>	The script description.
<name>	The script name
<oid>	The object identifier.

getScriptLog

Use this request to get a log of a script from your unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getScriptLog>
         <servicePass>
            <password></password>
            <userID>admin</userID>
         </servicePass>
         <devId>4755</devId>
         <serialNumber>FGT60C3G06500185</serialNumber>
         <logId>log1</logId>
   </r20:getScriptLog>
</soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<scriptname>	The name of the script log.
<devId>	The device ID. This is the primary device identifier.
<serialNumber>	The serial number of device.
<logId>	The log ID number.

The response indicates if the request was successful or if it failed.

Example response:

If the task is successful, you will get the log, if not, you will get an error message.

Request Field	Description
<errorMsg>	Indicates if the request was successful or it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Retrieved script log script name successfully. Retrieved current script log successfully. • -101: You must assign the script name when log ID exists. • -102: The script does not exist. The script log does not exist. • -106: soap_new_ns3__scriptLog error. soap_new_std__string error.
<errorMsg>	

getScriptLogSummary

Use this request to get a summary of a script log from your unit.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:getScriptLogSummary>
   <servicePass>
         <password></password>
         <userID>admin</userID>
   </servicePass>
   <devId>4755</devId>
   <serialNumber>FGT60C3G06500185</serialNumber>
   <maxLogs>2</maxLogs>
   </r20:getScriptLogSummary>
</soapenv:Body>
```

Request Field	Description
<sevicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<devId>	The device ID. This is the primary device identifier.
<serialNumber>	Serial number of device.
<maxLogs>	The ID number of the log.

The response indicates if the request was successful or if it failed.

Example response:

If the task is successful, you will get a message stating that the summary was created successfully.

If the task is unsuccessful, you will get a message similar to the one below. The details will vary, depending on the error.

```
<faultcode>SOAP-ENV:Client</faultcode>
<faultstring>No script log</faultstring>
      <detail>
         <error xmlns="http://localhost/">No script log</error>
      </detail>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or it failed. The error message consists of the error code and detail.
<errorcode>	Error code and message details: • 0: No script log. Retrieved device ID script log summary total number of records) successfully. • -104: Get script log summary failed. Failed to get any script log. • -106: Malloc memory failure; maybe caused by too many logs. `soap_new_ns3__scriptLogSummary` error
<errorMsg>	
<faultcode>	These are considered as generic SOAP errors. But there are cases that errors from the application level also return inside `<SOAP-ENV:Fault>` envelop. These errors are free-style, there are no error codes associated.
<faultstring>	

installConfig

When you have made configuration changes on the global or device database with your scripts, use this request to install the changes to the devices.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:installConfig>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <from>?</from>
         <!--Optional:-->
         <to>?</to>
         <!--Optional:-->
```

```
            <adom>?</adom>
            <!--Optional:-->
            <pkgoid>?</pkgoid>
            <!--Optional:-->
            <devId>?</devId>
            <!--Optional:-->
            <serialNumber>?</serialNumber>
            <!--Optional:-->
            <newRevName>?</newRevName>
            <instValidate>?</instValidate>
      </r20:installConfig>
   </soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<from>	The source of the configuration. The possible values are: global or local.
<to>	The destination of the configuration. The possible values are: local or remote.
<adom>	If the ADOM field is blank, the default ADOM will be that of the administrative user. If this administrator binds to all ADOMs, then the ADOM is root.
<pkgoid>	The package object identifier for a policy package.
<devId>	The device ID. This is the primary device identifier.
<serialNumber>	Serial number of device. This device identifier is secondary to the device ID.
<newRevName>	The new revision name.
<instValidate>	Pre-install policy verification.

If the installation is successful, the response is empty. Otherwise, web services returns an error message.

Example response: (updated configuration installed successfully)

```
<ns3:installConfigResponse/>
```

Example response: (updated configuration could not be installed)

```
<faultcode>SOAP-ENV:Client</faultcode>
<faultstring>Run install+save on deviceId 109 failed</faultstring>
<detail>
<error xmlns="http://localhost/">Run install+save on deviceId 109 failed</error>
```

```
</detail>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Read task ID to get configuration install result. • -101: Invalid ADOM. Unknown or unsupported device. The new revision name length should be 1 - 49 characters. Unsupported copy from GMS global to local, global, or remote. Unsupported copy action. • -102: The ADOM is locked. The device is in backup mode. Cannot create a temp file. Cannot connect with the target device ID. Cannot handle temp file. Not enough memory. Cannot get ADOM information. The package OID is invalid. Unknown or unsupported ADOM mode. • -104: Run install and save on device ID failed. Copy GMS global configurations to local failed. Copy and install GMS global configurations failed.
<errorMsg>	
<faultcode>	These are considered as generic SOAP errors. But there are cases that errors from application level also return inside `<SOAP-ENV:Fault>` envelop. These errors are free-style, there are no error codes associated.
<faultstring>	

runScript

Use this request to run a script. You can run a script on the global database, the device database, or on the managed device.

Example request:

```
<soapenv:Header/>
<soapenv:Body>
   <r20:runScript>
         <!--Optional:-->
         <servicePass>
            <!--Optional:-->
            <userID>?</userID>
            <!--Optional:-->
            <password>?</password>
         </servicePass>
         <!--Optional:-->
         <isGlobal>?</isGlobal>
         <!--Optional:-->
         <name>?</name>
         <!--Optional:-->
         <type>?</type>
         <!--Optional:-->
         <devId>?</devId>
```

```
            <!--Optional:-->
            <serialNumber>?</serialNumber>
            <!--Optional:-->
            <runOnDB>?</runOnDB>
            <!--Optional:-->
            <pkgoid>?</pkgoid>
      </r20:runScript>
   </soapenv:Body>
```

Request Field	Description
<servicePass>	XML structure consists of username and password variables.
<userID>	The administrator user name.
<password>	Administrator password options: Enter the administrator password or leave field blank for no password.
<isGlobal>	Set for global script.
<name>	The name of the script to run.
<type>	The script type, e.g. CLI, TCL, CLIGROUP.
<devId>	Provide the device ID when you run a script on the device or device database. You can also omit the device ID field and use the serial number to identify the unit. Set device ID to -1 when you run the script on the global database.
<serialNumber>	Serial number of device, FGT60C3G06500185, for example. This device identifier is secondary to device ID.
<runOnDB>	Run on database options: • 1: Run on global or device database, depending on the device ID. • 0: Run on the device. Specify the device ID or the serial number.
<pkgoid>	The package object identifier for a policy package.

If the script runs successfully, the response is empty. Otherwise, web services returns an error message.

Example response: (script ran successfully)

```
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
   <ns3:runScriptResponse>
         <errorMsg>
            <errorCode>0</errorCode>
            <errorMsg>Read taskId value to get runScript result</errorMsg>
         </errorMsg>
         <taskId>6</taskId>
   </ns3:runScriptResponse>
</SOAP-ENV:Body>
```

Response Field	Description
<errorMsg>	Indicates if the request was successful or if it failed. The error message consists of the error code and detail.
<errorCode>	Error code and message details: • 0: Read task ID value to get run script result. • -101: Invalid ADOM. No script name is assigned. Script type should be ALL,CLI,TCL or CLIGROUP. TCL script cannot run on database. Total number of devices or groups exceeds the limit. • -102: The ADOM is locked. Cannot find script in ALL, CLI, TCL, CLIGROUP set. Cannot get script type. Script OS type is invalid. Script ADOM is invalid. Cannot find valid admin login. User does not have permission to access script. The device ID/group ID does not belong to script ADOM. Please input a valid package name. • -104: Run script on device ID failed.
<errorMsg>	
<taskId>	Indicates the task ID number. If the `<waitTask>` was false, then the task ID is displayed.